Acorn Archimedes 500 series
Acorn R200 series
Technical Reference Manual

The products described in this manual, and products for use with it are subject to continuous development and improvement. All information of a technical nature and particulars of the products and their use (including the information and particulars in this manual) are given by Acorn Computers Limited in good faith. However, Acorn Computers Limited cannot accept any liability for any loss or damage arising from the use of any information or particulars in this manual, or any incorrect use of the products. All maintenance and service on the products must be carried out by Acorn Computers' authorised dealers or Approved Service Centres. Acorn Computers Limited can accept no liability whatsoever for any loss or damage caused by service, maintenance or repair by unauthorised personnel.

All correspondence should be addressed to:
The Customer Service department
Acorn Computers Limited
Fulbourn Road
Cherry Hinton
Cambridge CB1 4JN
United Kingdom

Support and information can also be obtained from the Acorn Support Information Database (SID). This is a Viewdata system available to registered SID users. Initially, access SID by dialling directly into the Guest User access (parity 7E1, speed V21/22/23/22bis,UK telephone number (0223) 243642): this will allow you to inspect the system and use a response frame for registration. Alternatively, access SID via Prestel (type *SID#): this will automatically register you on your first call.

Published by Acorn Computers Limited
ISBN 1 85250 086 7
Part number 0486,052
Issue 1, November 1990

Contents

About this manual

This manual is intended as a hardware reference for the following models:

- Archimedes 540
- Acorn R260
- Acorn R225

Throughout the remainder of this manual, the generic term *workstation* will be used to refer to the above, unless a reference to a specific model is required.

This manual supplements the basic information given on system hardware in the *Installation Guide*, supplied with certain models (also available for separate purchase).

This manual will be of interest to system integrators, software developers and those developing expansion cards for the workstation.

The operating systems, RISC OS and RISC iX, are covered at the user level in the *RISC OS User Guide* and the *RISC iX User Guide*, supplied with certain models (also available for separate purchase). Programmers and users requiring a greater depth of information about RISC OS and RISC iX will also need the following manuals:

- *RISC OS Programmer's Reference Manual* (4 volume set)
- *RISC iX Programmer's Reference Manual* (2 volume set).

They are available from Acorn authorised dealers.

Full details on the Acorn ARM chip set used in the workstation are given in the *Acorn RISC Machine (ARM) Family Data Manual, ISBN 0-13-781618-9*, available from:

VLSI Technology, Inc.
Application Specific Logic Products Division
8375 South River Parkway
Tempe, AZ 85284
USA
602-752-8574

or from the VLSI national distributor.

Note: This manual describes various PCB assemblies. The issue of each PCB is as defined by the relevant schematic.

Part 1 - System description

Introduction

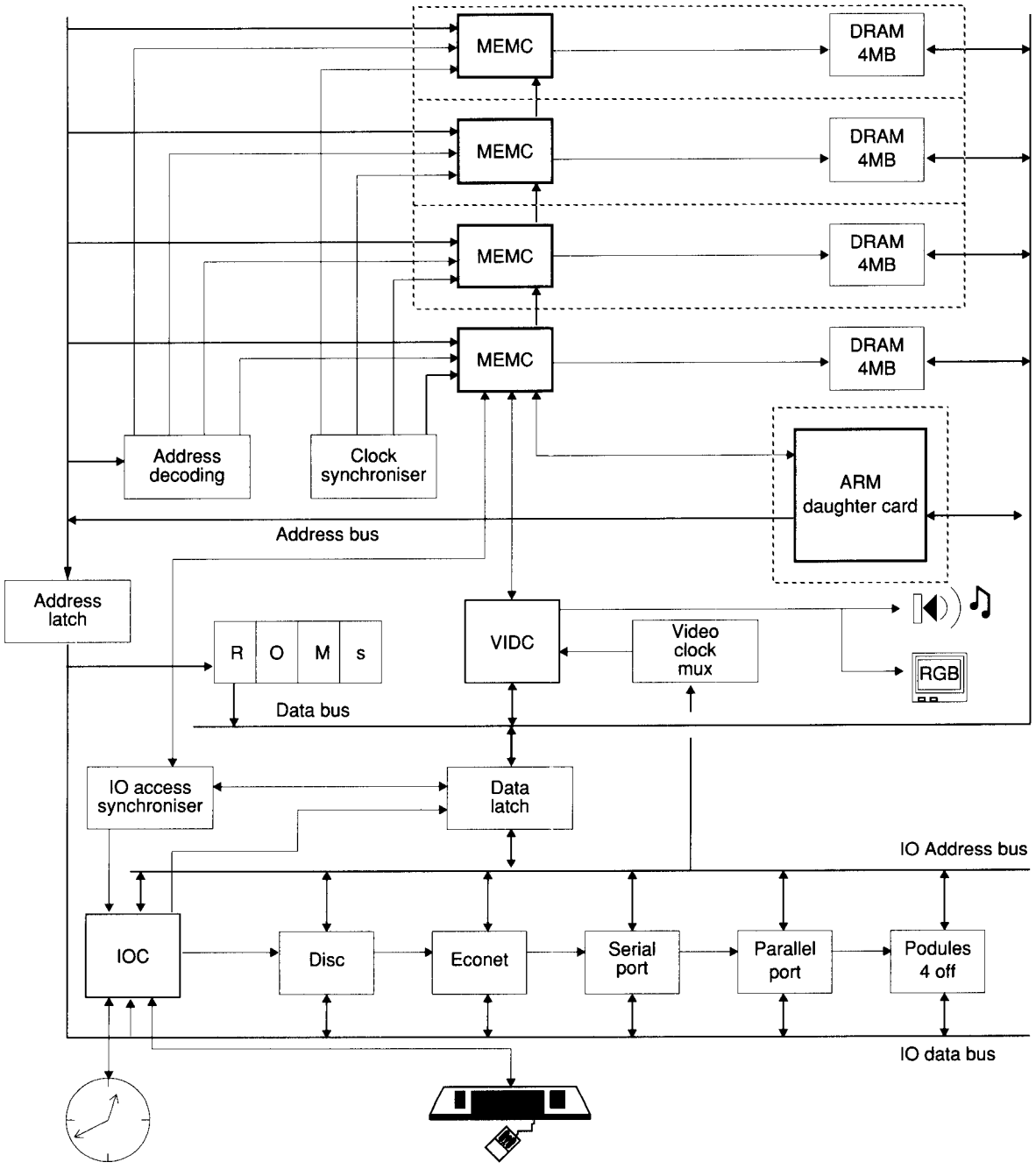
The workstation is built around the ARM chip set, comprising the Acorn RISC Machine (ARM) itself, the Memory Controller (MEMC), Video Controller (VIDC) and Input Output Controller (IOC).

The ARM CPU is fitted on a daughter card. Additionally, memory expansion cards are available, each with 4MB of RAM and a MEMC controller.

A block diagram of the workstation is shown below:

General

The ARM3 CPU is a pipelined, 32-bit reduced instruction set microprocessor which accepts instructions and manipulates data via a high speed 32-bit data bus and 26-bit address bus, giving a 64 MB uniform address space. The ARM supports virtual memory systems using a simple but powerful instruction set with good high-level language compiler support. The ARM3 version has 4KB of on-chip cache memory, which greatly increases the speed with which some data is handled (typically 2 - 3 times faster than ARM2).



MEMC acts as the interface between the ARM, VIDC, IOC, ROM (Read-Only Memory) and DRAM (Dynamic RAM) devices, providing all the critical system timing signals, including processor clocks.

Up to 4 MB of DRAM is connected to the 'Master' MEMC which provides all signals and refresh operations. A Logical to Physical Translator maps the Physical Memory into a 32 MB Logical address space (with three levels of protection) allowing Virtual Memory and Multi-Tasking operations to be implemented. Fast page mode DRAM accesses are used to maximise memory bandwidth. VIDC requests data from the RAM when required and buffers it in one of three FIFOs before using it. Data is requested in blocks of four 32-bit words, allowing efficient use of paged-mode DRAM without locking the system data bus for long periods.

MEMC supports Direct Memory Access (DMA) operations with a set of programmable DMA Address Generators which provide a circular buffer for Video data, a linear buffer for Cursor data and a double buffer for Sound data.

IOC controls the I/O bus and expansion cards, and provides basic functions such as the keyboard interface, system timers, interrupt masks and control registers. It supports a number of different peripheral cycles and all I/O accesses are memory mapped.

VIDC takes video data from memory under DMA control, serialises it and passes it through a colour look-up palette and converts it to analogue signals for driving the CRT guns. VIDC also controls all the display timing parameters and controls the position and pattern of the cursor sprite. In addition, it incorporates an exponential Digital to Analogue Converter (DAC) and stereo image table for the generation of high-quality sound from data in the DRAM.

VIDC is a highly programmable device, offering a very wide choice of display formats. The colour look-up palette which drives the three on-chip DACs is 13 bits wide, offering a choice from 4096 colours or an external video source.

The cursor sprite is 32 pixels wide and any number of rasters high. Three simultaneous colours (again from a choice of 4096) are supported and any pixel can be defined as transparent, making possible cursors of many shapes. It can be positioned anywhere on the screen. The sound system implemented on the device can support up to eight channels, each with a separate stereo position.

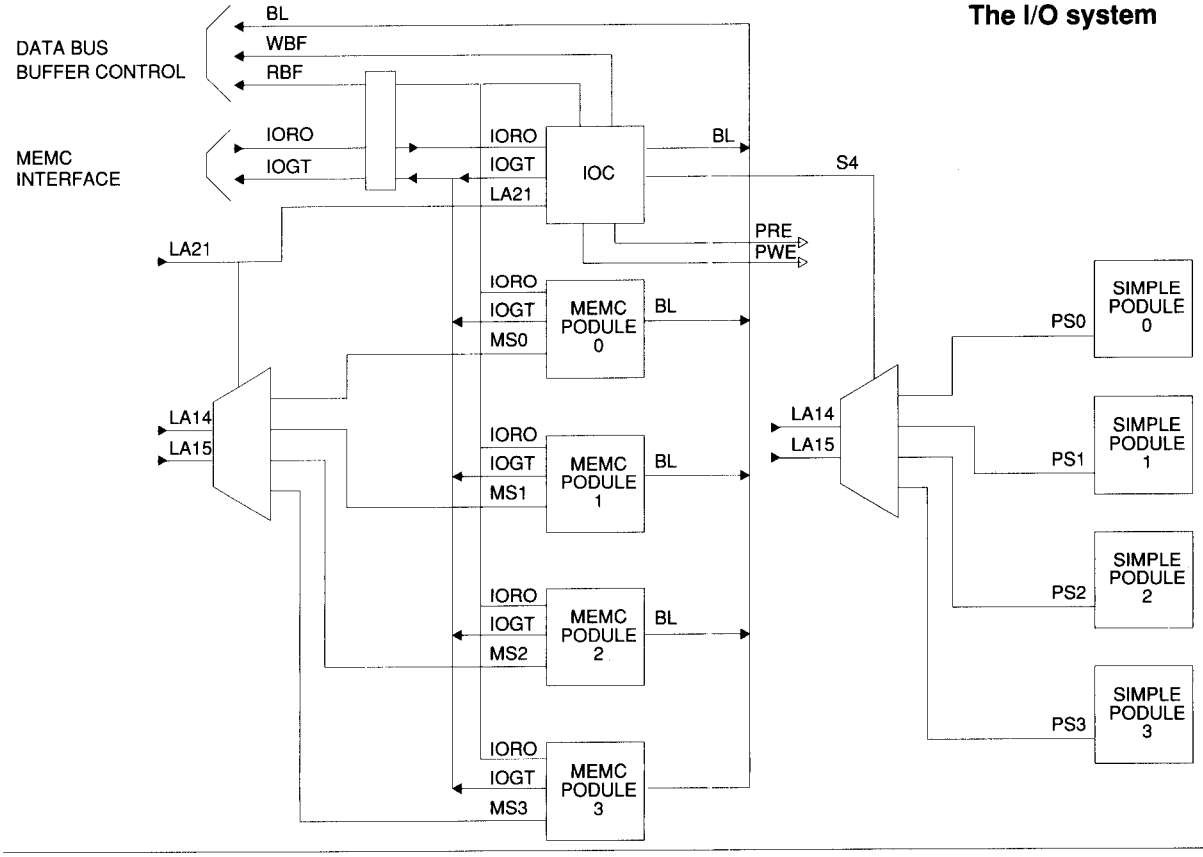
Additional memory is provided on daughter cards, in 4MB blocks. Each 4MB block is controlled by a separate MEMC.

NOTE: MEMCs **must** be Acorn Part Number 2201,393, to ensure correct timing parameters.

The I/O system

The I/O system is controlled by IOC, MEMC and two PALs. The I/O bus supports all the internal peripherals and the expansion cards.

The I/O system



This section is intended to give the reader a general understanding of the I/O system and should not be used to program the I/O system directly. The implementation details are liable to change at any time and only the published software interfaces should be used to manipulate the I/O system. Future systems may have a different implementation of the I/O system, and in particular the addresses (and number) of expansion card locations may move. For this reason, and to ensure that any device may be plugged into any slot, all driver code for expansion cards must be relocatable. References to the direct expansion card addresses should never be used. It is up to the machine operating system, in conjunction with the expansion card ID, to determine the address at which an expansion card should be accessed. To this extent, some of the following sections are for background information only.

System architecture

The I/O system (which includes expansion card devices) consists of a 16-bit data bus (BD[0:15]), a buffered address bus (LA[2:21]), and various control and timing signals. The I/O data bus is independent of the main 32-bit system data bus, being separated from it by bidirectional latches and buffers. In this way the I/O data bus can run at much slower speeds than the main system bus to cater for slower peripheral devices. The latches between the two buses, and hence the I/O bus timing, are controlled by the I/O controller, IOC. IOC caters for four different cycle speeds (slow, medium, fast and synchronous).

A typical I/O system is shown in the diagram on the previous page. For clarity, the data and address buses are omitted from this diagram.

System memory map

The system memory map is defined by master MEMC and the master PAL, and is shown below. Note that all system components, including I/O devices, are memory mapped.

I/O space memory map

This IOC-controlled space has allocation for simple expansion cards and MEMC expansion cards.

Data bus mapping

The I/O data bus is 16 bits wide. Bytewide accesses are used for 8-bit peripherals. The I/O data bus (BD[0:15]) connects to the main system data bus (D[0:31]) via a set of bidirectional data latches.

The mapping of the BD[0:15] bus onto the D[0:31] bus is as follows:

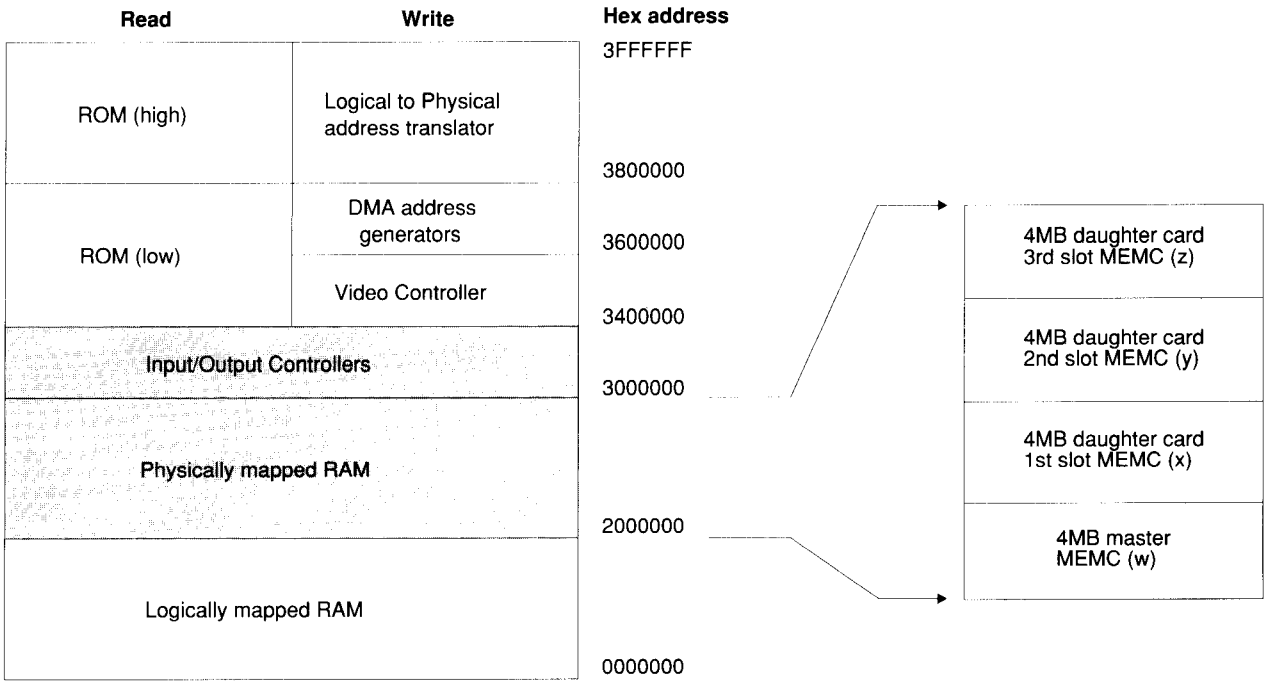
During a WRITE (ie ARM to peripheral) D[16:31] is mapped toBD[0:15].

During a READ (ie peripheral to ARM) BD[0:15] is mapped to D[0:15].

Byte accesses

Byte instructions are used to access bytewide expansion cards. A byte store instruction places the written byte on all four bytes of the word, and so correctly places the desired value on the lowest byte of the I/O bus. A byte or word load may be used to read a bytewide expansion card into the lowest byte of an ARM register.

System memory map



Half-word accesses

To access a 16-bit wide expansion card, half-word instructions are used. When storing, the half-word is placed on the upper 16 bits, D[16:31]. To maintain upwards compatibility with future machines, half-word stores replicate the written data on the lower half-word, D[0:15]. When reading, the upper 16 bits are undefined.

Expansion card identification

It is important that the system is able to identify what expansion cards (if any) are present, and where they are. This is done by reading the Podule (expansion card) Identification (PI) byte, or bytes, from the Podule Identification Field.

I/O address memory mapping

All I/O accesses are memory mapped. IOC is connected as detailed in this table:

IOC	ARM
OE	LA[21]
T[1]	LA[20]
T[0]	LA[19]
B[2]	LA[18]
B[1]	LA[17]
B[0]	LA[16]

Internal register memory map

Address	Read	Write
3200000H	Control	Control
3200004H	Serial Rx Data	Serial Tx Data
3200008H	-	-
320000CH	-	-
3200010H	IRQ status A	-
3200014H	IRQ request A	IRQ clear
3200018H	IRQ mask A	IRQ mask A
320001CH	-	-
3200020H	IRQ status B	-
3200024H	IRQ request B	-
3200028H	IRQ mask B	IRQ mask B
320002CH	-	-
3200030H	FIQ status	-
3200034H	FIQ request	-
3200038H	FIQ mask	FIQ mask
320003CH	-	-
3200040H	T0 count Low	T0 latch Low
3200044H	T0 count High	T0 latch High
3200048H	-	T0 go command
320004CH	-	T0 latch command
3200050H	T1 count Low	T1 latch Low
3200054H	T1 count High	T1 latch High
3200058H	-	T1 go command
320005CH	-	T1 latch command
3200060H	T2 count Low	T2 latch Low
3200064H	T2 count High	T2 latch High
3200068H	-	T2 go command
320006CH	-	T2 latch command
3200070H	T3 count Low	T3 latch Low
3200074H	T3 count High	T3 latch High
3200078H	-	T3 go command
320007CH	-	T3 latch command

Peripheral address

Cycle type	Bk	Base address	IC	Use
Fast	1	&3310000	1772	Floppy disc controller
Sync	2	&33A0000	6854	Econet controller *
Sync	3	&33B0000	6551	Serial line controller
Slow	4	&3240000	Podule 0	Expansion slot
Med	4	&32C0000	Podule 0	Expansion slot
Fast	4	&3340000	Podule 0	Expansion slot
Sync	4	&33C0000	Podule 0	Expansion slot
Slow	4	&3244000	Podule 1	Expansion slot
Med	4	&32C4000	Podule 1	Expansion slot
Fast	4	&3344000	Podule 1	Expansion slot
Sync	4	&33C4000	Podule 1	Expansion slot
Slow	4	&3248000	Podule 2	Expansion slot
Med	4	&32C8000	Podule 2	Expansion slot
Fast	4	&3348000	Podule 2	Expansion slot
Sync	4	&33C8000	Podule 2	Expansion slot
Slow	4	&324C000	Podule 3	Expansion slot
Med	4	&32CC000	Podule 3	Expansion slot
Fast	4	&334C000	Podule 3	Expansion slot
Sync	4	&33CC000	Podule 3	Expansion slot
Fast	5	&335000	LS374	Printer Data
Fast	5	&3350018	HC574	Latch B (See next page for details)
Fast	5	&3350040	HC574	Latch A (See next page for details)
Fast	5	&3350048	HC175	Latch C (See next page for details)
Fast	6	&3360000	16L8	Podule interrupt request register
Fast	6	&3360004	16L8	Podule interrupt mask register
Slow	7	&3270000		Extended external podule space

*if fitted

I/O programming details

External latch A

External latch A is a write only latch used to control parts of the floppy disc sub-system:

Bit	Name	Function
0-3	Floppy disc sel.	These bits select the floppy disc drive 0 through 3 when written LOW. Only one bit should be LOW at any one time.
4	Side select	This controls the side select line of the floppy disc interface. 0 = Side 1 (upper) 1 = Side 0 (lower)
5	Floppy motor	This bit controls the floppy disc on/off control motor line. Its exact use depends on the type of drive.
6	In Use	This bit controls the IN USE line of the floppy disc. Its exact use depends on the type of drive.
7		Not used.

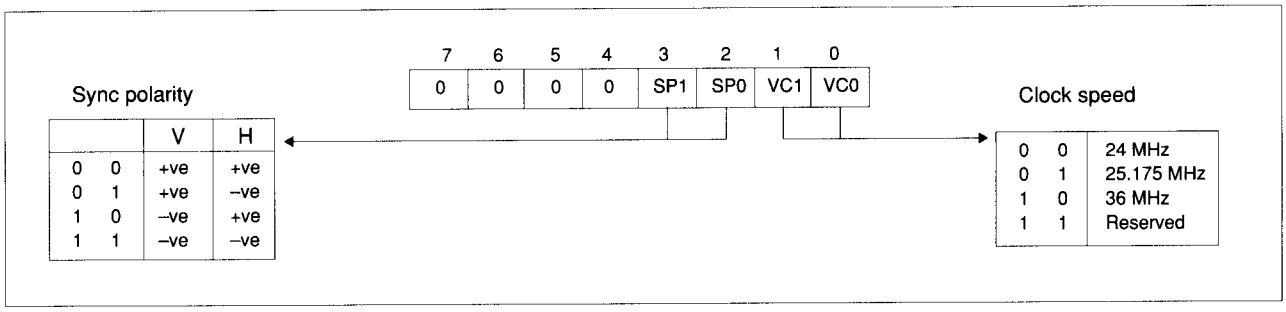
External latch B

External Latch B is a write only register shared between several users who must maintain a consistent RAM copy. Updates must be made with IRQ disabled.

Bit	Name	Function
0-2		CD[0:2] should be programmed CD[0.2] LOW for future compatibility. CD[1] controls the floppy disc data separator format. CD[1] = 0 Double Density CD[1] = 1 Single Density
3	FDCR	This controls the floppy disc controller reset line. When programmed LOW, the controller is RESET.
4	Printer Strobe	This is used to indicate valid data on the printer outputs. It should be set HIGH when valid data has been written to the printer port and LOW after about 5
[5:6]	AUX [1:2]	Not used.
7	HS3	Not used.

External latch C

External latch C is a write only register that is used to control video sync polarity and clock speed.



Interrupts

The I/O system generates two independent interrupt requests, IRQ and FIQ. Interrupt requests can be caused by events internal to IOC or by external events on the interrupt or control port input pins.

The interrupts are controlled by four types of register:

- status
- mask
- request
- clear

The status registers reflect the current state of the various interrupt sources. The mask registers determine which sources may generate an interrupt. The request registers are the logical AND of the status and mask registers and indicate which sources are generating interrupt requests to the processor. The clear register allows clearing of interrupt requests where appropriate. The mask registers are undefined after power up.

The IRQ events are split into two sets of registers, A and B. There is no priority encoding of the sources.

Internal Interrupt Events

- Timer interrupts TM[0:1]
- Power-on reset POR
- Keyboard Rx data available SRx
- Keyboard Tx data register empty STx
- Force interrupts 1.

External Interrupt Events

- IRQ active low inputs IL[0:7] wired as (0-7 respectively) PFIQ, SIRQ, SLC1, not used, DCIRQ, PIRQ, PBSY and RII.
- IRQ falling-edge input IF wired as PACK
- IRQ rising-edge input IR wired as VFLY
- FIQ active high inputs FII[0:1] wired as FFDQ and FFIQ
- FIQ active low input FL wired as EFIQ
- Control port inputs C[3:5].

Podule interrupt mask

Podule IRQ can be masked by writing a 0 to the Podule IRQ mask register at &3360004. This will disable the interrupt.

The request register at &3360000 is a logical AND of Podule IRQ and the mask register, ie it is1 if Podule IRQ is not masked.

IRQ status A

Bit	Name	Function
0	PBSY	This bit indicates that the printer is busy.
1	RI	This bit indicates that a Ringing Indication has been detected by the serial line interface.
2	Printer Ack	This bit indicates that a printer acknowledgement bit has been received.
3	Vertl Flyback	This bit indicates that a vertical flyback has commenced.
4	Power-on reset	This bit indicates that a power-on reset has occurred.
[5:6]	Timer 0 and Timer 1 events	These bits indicate that events have occurred. Note: latched interrupt.
7	Force	This bit is used to force an IRQ request. It is usually owned by the FIQ owner

IRQ status B

Bit	Name	Function
0	Podule FIQ req	This bit indicates that a Podule FIQ request has been received. It should usually be masked OFF.
1	Snd buffr swap	This bit indicates that the MEMC sound buffer pointer has been relocated.
2	Serial line ctrlr	This bit indicates that 65C51 serial line controller interrupt has occurred.
3	H disc interrupt	This bit indicates that a hard disc interrupt has occurred.
4	Disc changed	This bit indicates that the floppy disc interrupthas been removed.
5	Pod. interr req	This bit indicates that a Podule IRQ request has occurred.
6	Keyb Tx event	This bit indicates that the keyboard transmit register is empty and may be reloaded.
7	Keybd Rx event	This bit indicates that the keyboard reception register is full and may be read.

Interrupt status FIQ

Bit	Name	Function
0	Floppy disc data request	This bit indicates that a floppy disc Data Request has occurred.
1	Floppy disc interrupt request	This bit indicates that a floppy disc Interrupt Request has occurred.
2	Econet Interrupt request	This bit indicates that an Econet Interrupt Request has occurred.
3-5	C[3:5]	See IOC data sheet for details.
6	Podule FIQ req	This bit indicates that a podule FIQ Request has occurred.
7	Force	This bit allows an FIQ Interrupt Request to be generated.

Control port

The control register allows the external control pins C[0:5] to be read and written and the status of the PACK and VFLY inputs to be inspected. The C[0:5] bits manipulate the C[0:5] I/O port. When read, they reflect the current state of these pins. When written LOW the output pin is driven LOW. These outputs are open-drain, and if programmed HIGH the pin is undriven and may be treated as an input.

On reset all bits in the control register are set to 1.

Bit	Name	Function
C[7]	VFLYBK	Allows the state of the (VFLYBK) and Test Mode signal to be inspected. This bit will be read HIGH during vertical flyback and LOW during display. See VIDC datasheet for details. This bit MUST be programmed HIGH to select normal operation of the chip.
C[6]	PACK & Test Mode	Allows the state of the parallel printer acknowledge input to be inspected. This bit MUST be programmed HIGH to select normal operation of the chip.
C[5]	SMUTE	This controls the muting of the internal speaker. It is programmed HIGH to mute the speaker and LOW to enable it. The speaker is muted on reset.
C[4]		Available on the Auxiliary I/O connector.
C[3]		Programmed HIGH, unless Reset Mask is required.
C[2]	READY	Used as the floppy disc (READY) i nput and must be programmed HIGH.
C[1:0]	SDA, SCL	The C[0:1] pins are used to implement the I2C bus the bi-directional serial I2C bus to which the Real Time Clock

The sound system

The sound system is based on the VIDC stereo sound hardware. External analogue anti-alias filters are used which are optimised for a 20 kHz sample rate. The high quality sound output is available from a 3.5mm stereo jack socket at the rear of the machine which will directly drive personal stereo headphones or alternatively an amplifier and speakers. One internal speaker is fitted, to provide mono audio.

VIDC sound system hardware

VIDC contains an independent sound channel consisting of the following components: A four-word FIFO buffers 16 8-bit sound samples with a DMA request issued whenever the last byte is consumed from the FIFO. The sample bytes are read out at a constant sample rate programmed into the 8-bit Audio Frequency Register. This may be programmed to allow samples to be output synchronously at any integer value between 3 and 255 microsecond intervals.

The sample data bytes are treated as sign plus 7-bit logarithmic magnitude and, after exponential digital to analogue conversion, de-glitching and sign-bit steering, are output as a current at one of the audio output pins to be integrated and filtered externally.

VIDC also contains a bank of eight stereo image position registers each of three bits. These eight registers are sequenced through at the sample rate with the first register synchronised to the first byte clocked out of the FIFO. Every sample time is divided into eight time slots and the 3-bit image value programmed for each register is used to pulse width modulate the output amplitude between the LEFT and RIGHT audio current outputs in multiples of time slot subdivisions. This allows the signal to be spatially positioned in one of seven stereo image positions.

MEMC sound system hardware

MEMC provides three internal DMA address registers to support Sound buffer output; these control the DMA operations performed following Sound DMA requests from VIDC. The registers allow the physical addresses for the START, PNTR (incremental) and END buffer pointers to a block of data in the lowest half Megabyte of physical RAM to be accessed. These operate as follows: programming a 19-bit address into the PNTR register sets the physical address from which sequential DMA reads will occur (in multiples of four words), and programming the END pointer sets the last physical address of the buffer. Whenever the PNTR register increments up to this END value the address programmed into the START register is automatically written into the PNTR register for the DMA to continue with a new sample buffer in memory.

A Sound Buffer Interrupt (SIRQ) signal is generated when the reload operation occurs which is processed by IOC as a maskable interrupt (IRQ) source.

MEMC also includes a sound channel enable/disable signal. Because this enable/disable control signal is not synchronised to the sound sampling, requests will normally be disabled after the waveforms which are being synthesised have been programmed to decay to zero amplitude; the last value loaded into the Audio data latch in the VIDC will be output to each of the Stereo image positions at the current Audio Sample rate.

IOC sound system hardware

IOC provides a programmed output control signal which is used to turn the internal speaker on or off, as well as an interrupt enable/status/reset register interface for the Sound Start Buffer reload signal generated by MEMC.

The internal speaker may be muted by the control line SMUTE which is driven from the IOC output C5. On reset this signal will be taken high and the internal speaker will be muted.

The stereo output to the headphone socket is not muted by SMUTE and will always reflect the current output of the DAC channels.

The keyboard and mouse

The keyboard assembly comprises a membrane keyswitch panel connected to an adaptor PCB, which serialises the keyboard and mouse data; connection to the ARM is made via a serial link to the IOC. The ARM reads and writes to the KART registers in the IOC. The protocol is essentially half duplex, so in normal operation the keyboard will not send a second byte until it has received an Ack. The only exception to this is during the reset protocol used to synchronise the handshaking, where each side is expecting specific responses from the other, and will not respond further until it has these.

In addition to this simple handshaking system, the keyboard will not send mouse data unless specifically allowed to, as indicated by Ack Mouse, which allows the transmission of one set of accumulated mouse coordinate changes, or the next move made by the mouse. While it is not allowed to send mouse changes, the keyboard will buffer mouse changes.

A similar handshake exists on key changes, transmitted as key up and key down, and enabled by Ack Scan. At the end of a keyboard packet (two bytes) the operating system will perform an Ack Scan as there is no protocol for re-enabling later. Mouse data may be requested later by means of Request Mouse Position (RQMP).

Key codes

The keyboard identifies each key by its row and column address in the keyboard matrix. Row and column codes are appended to the key up or down prefix to form the complete key code.

For example, Q key down – the complete row code is 11000010 (&C2) and the column code is 11000111 (&C7).

Note: Eight keys have N key roll over. The operating system is responsible for implementing two-key rollover, therefore the keyboard controller transmits all key changes (when enabled). The keyboard does not operate any auto-repeat; only one down code is sent, at the start of the key down period.

Data protocol

Data transmissions from the keyboard are either one or two bytes in length. Each byte sent by the keyboard is individually acknowledged. The keyboard will not transmit a byte until the previous byte has been acknowledged, unless it is the HRST (HardReSeT) code indicating that a power on or user reset occurred or that a protocol error occurred; see paragraph below.

Reset protocol

The keyboard restarts when it receives an HRST code from the ARM. To initiate a restart the keyboard sends an HRST code to the ARM, which will then send back HRST to command a restart.

The keyboard sends HRST to the ARM if:

- A power-on reset occurs
- A user reset occurs
- A protocol error is detected.

After sending HRST, the keyboard waits for an HRST code. Any non-HRST code received causes the keyboard to resend HRST. The pseudo program on this page illustrates the reset sequence or protocol.

Note, the on/off state of the LEDs does not change across a reset event, hence the LED state is not defined at power on. The ARM is always responsible for selecting

```
START reset
ONerror Send HRST code to ARM then wait for code from ARM.
IF code = HRST THEN restart ELSE error
ONrestart   clear mouse position counters
            set mouse mode to data only in response to an RMPS request.
            stop key matrix scanning and set key flags to up
            send HRST code to ARM
Wait for next code
IF code = RAK1 THEN send RAK1 to ARM    ELSE   error
Wait for next code
IF code = RAK2 THEN send RAK2 to ARM    ELSE   error
Wait for next code
IF code = SMAK THEN mouse mode to send if not zero and enable key scan
ELSE  IF code = SACK THEN enable key scanning
ELSE  IF code = MACK THEN set mouse mode to send when not zero
ELSE  IF code = NACK THEN do nothing    ELSE   error
END reset
```

Reset sequencing

Direction	Code	Expected reply	Action on wrong reply (Sender)	Action on timeout (Sender)	Action if unexpected (Receiver)
ARM -> Kb	Hard reset	Hard reset	Resend	Resend	Hard reset
Kb -> ARM	Hard reset	Reset Ack 1	Resend	Nothing	Hard reset
ARM -> Kb	Reset Ack 1	Reset Ack 1	Hard reset	Hard reset	Hard reset
Kb -> ARM	Reset Ack 1	Reset Ack 2	Nothing	Nothing	Hard reset
ARM -> Kb	Reset Ack 2	Reset Ack 2	Hard reset	Hard reset	Hard reset

the LED status. After the reset sequence, key scanning will only be enabled if a scan enable acknowledged (SACK or SMAK) was received from the ARM.

Data transmission

When enabled for scanning, the keyboard controller informs the ARM of any new key down or new key up by sending a two byte code incorporating the key row and column addresses. The first byte gives the row and is acknowledged by a byte acknowledge (BACK) code from the ARM. If BACK was not the acknowledge code then the error process (ON error) is entered. If the BACK code was received, the keyboard controller sends the column information and waits for an acknowledge. If either a NACK, SACK, MACK or SMAK acknowledge code is received, the keyboard controller continues by processing the ACK type and selecting the mouse and scan modes implied. If the character received as the second byte acknowledge was not one of NACK/MACK/SACK/SMAK then the error process is entered.

Mouse data

Mouse data is sent by the keyboard controller if requested by a RQMP request from the ARM or if a SMAK or MACK has enabled transmission of non-zero values. Two bytes are used for mouse position data. Byte one encodes the accumulated movement along the X axis while byte two gives Y axis movement.

Both X and Y counts must be transferred to temporary registers when data transmission is triggered, so that accumulation of further mouse movement can occur. The X and Y counters are cleared upon each transfer to the transmit holding registers. Therefore, the count values are relative to the last values sent. The ARM acknowledges the first byte (Xcount) with a BACK code and the second byte (Ycount) with any of NACK/MACK/SACK/SMAK. A protocol failure causes the keyboard controller to enter the error process (ON error).

When transmission of non-zero mouse data is enabled, the keyboard controller gives key data transmission priority over mouse data except when the mouse counter over/underflows.

Acknowledge codes

There are seven acknowledge codes which may be sent by the ARM. RAK1 and RAK2 are used during the reset sequence. BACK is the acknowledge to the first byte of a 2-byte keyboard data set. The four remaining types, NACK/MACK/SACK and SMAK, acknowledge the final byte of a data set. NACK disables key scanning and therefore key up/down data transmission as well as setting the mouse mode to send data only on RQMP request. SACK enables key scanning and key data transmission but disables unsolicited mouse data. MACK disables key scanning and key data transmission and enables the transmission of mouse count values if either X or Y counts are non-zero. SMAK enables key scanning and both key and mouse data transmission. It combines the enable function of SACK and MACK.

While key scanning is suspended (after NACK or MACK) any new key depression is ignored and will not result in a key down transmission unless the key remains down after scanning resumes following a SACK or SMAK.

Code values

Mnemonic	msb	lsb	Comments
HRST	1111	1111	1-byte command, keyboard reset.
RAK1	1111	1110	1-byte response in reset protocol.
RAK2	1111	1101	1-byte response in reset protocol.
RQPD	0100	xxxx	1-byte from ARM, encodes four bits of data.
PDAT	1110	xxxx	1-byte from keyboard, echoes four data bits of RQPD.
RQID	0010	0000	1-byte ARM request for keyboard ID.
KBID	10xx	xxxx	1-byte from keyboard encoding keyboard ID.
KDDA	1100	xxxx	New key down data. Encoded Row (first byte) and column (second byte) numbers.
KUDA	1101	xxxx	Encoded Row (first byte) and column (second byte) numbers for a new key up.
RQMP	0010	0010	1-byte ARM request for mouse data.
MDAT	0xxx	xxxx	Encoded mouse count, X (byte1) then Y (byte2). Only from ARM to keyboard.
BACK	0011	1111	Ack for first keyboard data byte pair.
NACK	0011	0000	Last data byte Ack, selects scan/mouse mode.
SACK	0011	0001	Last data byte Ack.
MACK	0011	0010	Last data byte Ack.
SMAK	0011	0011	Last data byte Ack.
LEDS	0000	0xxx	bit flag to turn LED(s) on/off.
PRST	0010	0001	From ARM, 1-byte command, does nothing.

x is a data bit in the Code; e.g. xxxx is a four bit data field

Similarly, a key release is ignored while scanning is off.

Commands may be received at any time. Therefore, commands can be interleaved with acknowledge replies from the ARM, eg keyboard sends KDDA (first byte), keyboard receives command, keyboard receives BACK, keyboard sends KDDA (second byte), keyboard receives command, keyboard receives SMACK. If the HRST command is received the keyboard immediately enters the restart sequence. The LEDS and PRST commands may be acted on immediately. Commands which require a response are held pending until the current data protocol is complete. Repeated commands only require a single response from the keyboard.

ARM commands

Mnemonic	Function
HRST	Reset keyboard.
LEDS	Turns key cap LEDs on/off. A three bit field indicates which state the LEDs should be in. Logic 1 is ON, logic 0 (zero) OFF. D0 controls CAPS LOCK D1 controls NUM LOCK D2 controls SCROLL LOCK
RQM	Request mouse position (X,Y counts).
RQID	Request keyboard identification code. The computer is manufactured with a 6-bit code to identify the keyboard type to the ARM. Upon receipt of RQID the keyboard controller transmits KBID to the ARM.
PRST	Reserved for future use, the keyboard controller currently ignores this command.
RQPD	For future use. The keyboard controller will encode the four data bits into the PDAT code data field and then send PDAT to the ARM.

Mouse interface

The mouse interface has three switch sense inputs and two quadrature encoded movement signals for each of the X axis and Y axis directions. Mouse key operations are debounced and then reported to the ARM using the Acorn key up / key down protocol. The mouse keys are allocated unused row and column codes within the main key matrix.

Switch 1 (left)	Row code - 7	Column code - 0
Switch 2 (middle)	Row code - 7	Column code - 1
Switch 3 (right)	Row code - 7	Column code - 2

For example, switch 1 release would give 11010111 (&D7) as the complete row code, followed by 11010000 (&D0) for the column code.

Note: Mouse keys are disabled by NACK and MACK acknowledge codes, and are only enabled by SACK and SMAK codes, ie they behave in the same way as the keyboard keys.

The mouse is powered from the computer 5V supply and may consume up to 100mA.

Movement signals

Each axis of movement is independently encoded in two quadrature signals. The two signals are labelled REFerence and DIRection (eg X REF and X DIR). The table below defines the absolute direction of movement. Circuitry in the keyboard decodes the quadrature signals and maintains a signed 7-bit count for each axis of mouse movement.

Initial state		Next state		
REF	DIR	REF	DIR	
1	1	1	0	Increase count by one for each change of state.
1	0	0	0	
0	0	0	1	
0	1	1	1	
1	1	0	1	Decrease count by one for each change of state.
0	1	0	0	
0	0	1	0	
1	0	1	1	

When count overflow or underflow occurs on either axis both X and Y axis counts lock and ignore further mouse movement until the current data has been sent to the ARM.

Overflow occurs when a counter holds its maximum positive count (0111111 binary). Underflow occurs when a counter holds its maximum negative count (1000000 binary).

Base Keyswitch mapping (UK 103 key keyboard)

Key size	Key name	Row code	Col. code	Notes
1	Esc	0	0	1
1	F1	0	1	2
1	F2	0	2	2
1	F3	0	3	2
1	F4	0	4	2
1	F5	0	5	2
1	F6	0	6	2
1	F7	0	7	2
1	F8	0	8	2
1	F9	0	9	2
1	F10	0	A	2
1	F11	0	B	2
1	F12	0	C	2
1	Print	0	D	1,3
1	Scroll	0	E	1
1	Break	0	F	1
1	~	1	0	
1	1	1	1	
1	2	1	2	
1	3	1	3	
1	4	1	4	
1	5	1	5	
1	6	1	6	
1	7	1	7	
1	8	1	8	
1	9	1	9	
1	0	1	A	
1	-_	1	B	
1	=+	1	C	
1	£¤	1	D	
1	Backspc	1	E	1
1	Insert	1	F	1
1	Home	2	0	1,3
1	Pgup	2	1	1
1	Numlock	2	2	1,4
1	/	2	3	1
1	*	2	4	1
1	#	2	5	1
1.5	Tab	2	6	1
1	Q	2	7	
1	W	2	8	
1	E	2	9	
1	R	2	A	
1	T	2	B	
1	Y	2	C	
1	U	2	D	
1	I	2	E	
1	O	2	F	
1	P	3	0	
1	[{	3	1	
1	]}	3	2	
1.5	\	3	3	
1	Delete	3	4	1
1	Copy	3	5	1
1	Pgdwn	3	6	1
1	7	3	7	
1	8	3	8	
1	9	3	9	
1	-	3	A	1
1.75	Ctrl	3	B	1,3
1	A	3	C	
1	S	3	D	
1	D	3	E	
1	F	3	F	
1	G	4	0	
1	H	4	1	
1	J	4	2	
1	K	4	3	
1	L	4	4	
1	;:	4	5	
1	'"	4	6	
2.25	Return	4	7	1
1	4	4	8	
1	5	4	9	
1	6	4	A	
1	+	4	B	1

Row and column codes are in hexadecimal.

Notes:
1 Key colour - dark grey.
2 Key colour - dark grey.
3 Key position with N key rollover.
4 Green LED under key cap.

Keyswitch mapping (cont.)

Key Size	Key Name	Row code	Col. code	Notes
2.25	shift	4	C	1,3
1	Z	4	E	
1	X	4	F	
1	C	5	0	
1	V	5	1	
1	B	5	2	
1	N	5	3	
1	M	5	4	
1	,<	5	5	
1	.>	5	6	
1	/	5	7	
2.75	shift	5	8	1,3
1	crsrUp	5	9	1
1	1	5	A	
1	2	5	B	
1	3	5	C	
1.5	Caps	5	D	1,4
1.5	Alt	5	E	1,3
7.0	Space	5	F	
1.5	Alt	6	0	1,3
1.5	Ctrl	6	1	1,3
1	crsrLt	6	2	1
1	crsrDn	6	3	1
1	crsrRt	6	4	1
2.0	0	6	5	
1	.	6	6	
2.0	Enter	6	7	1

Row and column codes are in hexadecimal.

Notes:
1. Key colour - dark grey.
2. Key colour - dark grey.
3. Key position with N key rollover.
4. Green LED under key cap.

Floppy disc drive

The floppy disc drive used on the workstations (except discless) is a one-inch high drive, taking 3.5 inch double-sided double-density floppy discs.

Performance

Capacity	1 MB (unformatted)
Track to track step rate	3ms
Seek settle time	15ms
Write to read timing	1200µs
Power-on to drive ready	1000ms
Power supply	+5Vdc (+/– 5%)
Maximum power	2 Watts (continuous)

Power connector

The power connector is a 4-pin, 2.5mm pitch type.

Pin	Signal
1	+5V
2	0V
3	0V
4	+12V

Interface connector

The interface connector is a 34-way, 2 row, 0.1 inch pitch type, with pinouts as shown below:

Pin		Signal	Dir
Retn	Signal		main (PCB)
1	2	Disc change	I
3	4	In use	I
5*	6	Drive select 3	O
7*	8	Index	I
9*	10	Drive select 0	O
11*	12	Drive select 1	O
13	14	Drive select 2	O
15	16	Motor ON	O
17	18	Direction	O
19	20	Step/Disc chg rst	O
21	22	Write data	O
23	24	Write gate	O
25	26	Track 0	I
27	28	Write protect	I
29	30	Read data	I
31	32	Side 1 select	O
33	34	Ready	I

*Optionally +5V I = Input O = Output

Power supply

Performance characteristics

Performance	Min	Nom	Max	Units
Input voltage (47-53 Hz)	198	220/ 240	264	Vac
*Input voltage (57-63 Hz)	98	110	132	Vac
Output voltage VO1	4.9	5	5.1	Vdc
Output current IO1	1.5	-	12.2	Amps dc
Output ripple and noise VO1			50	mV pk-pk BW 0-20MHz
Overshoot VO1			0.1	Vdc
Overvoltage prot VO1 (thrshld)	5.8	-	7.0	Vdc
Surge output current IO1	-	-	14.5	Amps dc
Surge output current duration	-	-	1.0	Sec
Output voltage VO2	11.4	12	12.6	Vdc
Output current IO2	0	-	3.2	Amps dc
Output ripple and noise VO2			100	mV pk-pk BW 0-20 MHz
Overshoot VO2			0.2	Vdc
Surge output current IO2	-	-	4	Amps dc
Surge output current duration	-	-	10.0	Sec
Output voltage VO3	-4.5	-5	-5.5	Vdc
Output current IO3	0	-	0.3	Amps dc
Output ripple and noise VO3			50	mV pk-pk BW 0-20MHz
Overshoot VO3			0.1	Vdc
Efficiency	63	-	-	%@max Id, nominal I/P volt
Total output power	-	-	100	Watts cont.
			122	Watts srge

*Manufacturing option

DANGER

THE POWER SUPPLY IS A SEPARATE REPLACEABLE MODULE, AND CONTAINS NO USER SERVICEABLE PARTS.

ALL ACORN POWER SUPPLIES CONTAIN HAZARDOUS VOLTAGES AND MUST NOT BE MODIFIED OR REPAIRED BY ANYONE OTHER THAN AUTHORISED ACORN SERVICE CENTRES.

POWER SUPPLY UINTS MAY ONLY BE FITTED BY AN AUTHORISED ACORN SERVICE CENTRE.

SAFETY EARTH CONTINUITY TESTING MUST BE CARRIED OUT WHEN ANY POWER SUPPLY IS FITTED.

Hard disc drive

The hard disc drive used on the workstations (except discless) is an internally-fitted SCSI device. For more information on the types of SCSI drive usable, see the *SCSI Expansion Card User Guide*.

Case colour specification

The colour of the cream plastic mouldings, the main case and the back panels which are painted, is Pearl White RAL 1013C.

The colour of the light grey front sub-moulding and the light grey keyboard keycaps is Pantone warm grey 3.

The colour of the darker grey keytops is Pantone warm grey 6.

Part 2 - Interface cards

Ethernet interface

Where an Ethernet interface is fitted, it is provided by one of two different types of Ethernet expansion card, identified as Ethernet I and Ethernet II. Both cards can support either a 'thick' or 'thin' (*Cheapernet*) Ethernet interface.

Overview

Ethernet was developed by the Xerox Corporation in the early 1970s and a specification made available in 1980. This specification known as the 'Blue Book' was used as the basis for the IEEE and ECMA standards. All new equipment (including this product) is or should be designed to the IEEE standard. This allows networking with existing Ethernet equipment, at least at the physical level.

An understanding of the basic architecture of the Ethernet/IEEE 802.3 standard is assumed. The Intel publication *The LAN Components User's Manual* is particularly useful and contains a suitable introduction to local area network standards. It is recommended that you obtain a copy if you require a wider understanding, as reference to it is made in this document.

Ethernet I expansion card

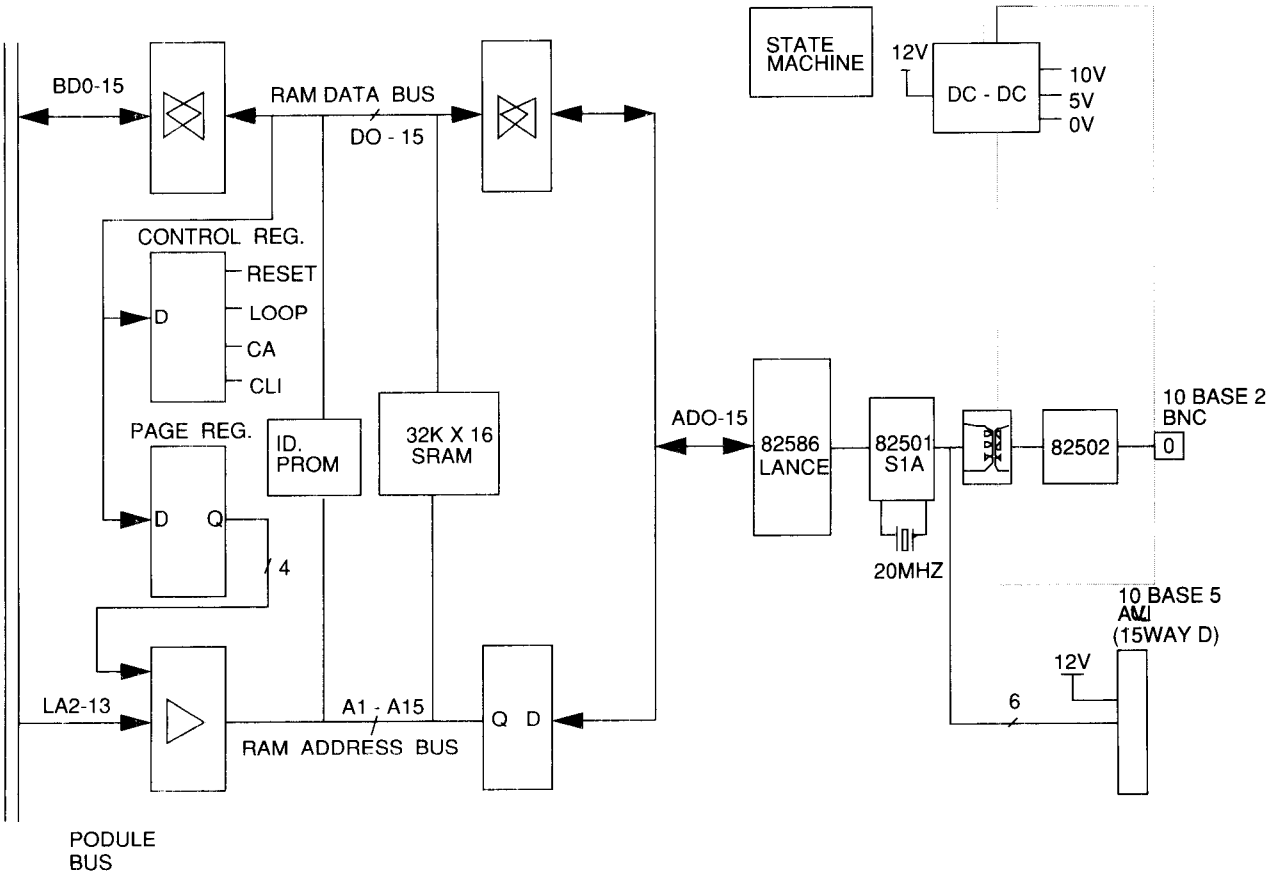
Basic operation and block diagram

The figure below is a block diagram of the Ethernet/Cheapernet podule.

The main functional blocks are:

- the net controller: Intel 82586 (LANCE)
- the serial interface adaptor: Intel 82501 (SIA)
- transceiver: Intel 82502
- attachment unit interface (AUI) socket (D-type)
- isolation transformers and power supply
- bus buffers and transceivers
- the RAM buffer
- the RAM page register
- a PROM based 'extended' podule ID
- the control register
- the PAL based state machine.

Figure 1: Ethernet I expansion card block diagram



The Intel chip set

As the Xerox and IEEE standards have become widely accepted, a number of systems companies have produced VLSI devices that considerably reduce the design effort required to implement a connection. The most notable of these are by Advanced Micro Devices (AMD) and Intel.

The Intel chip set comprising the 82586 local area network coprocessor, the 82501 Ethernet serial interface, and the 82502 Ethernet transceiver chip has been used in this design.

The 82586 and other similar local area network controllers are generally referred to by the acronym LANCE, even though this is a trademark of AMD.

The 82586 LANCE performs media access control, framing, pre/postamble generation and stripping, source address generation, CRC checking, and short packet detection. In addition diagnostic functions such as Time Domain Reflectometry (TDR) can be performed.

The 82501 serial interface adapter (SIA) performs Manchester encoding/decoding, receives clock recovery and directly drives the attachment unit interface (AUI) to the cable mounted Ethernet transceiver. In addition the 82501 operates a watchdog to prevent continuous transmission (a fault condition), and provides a loop-back test facility. A second source for this device is SEEQ who manufacturer a similar part, the DQ8023A. This part however is not identical and will not perform TDR correctly.

The 82502 transceiver applies transmit data to, and removes receive data from the Cheapernet cable interface. This devices performs a similar function to the cable mounted Ethernet transceiver.

The dual port memory

The LANCE is a true coprocessor and is designed to perform scatter-gather DMA. In common with other LANCE chips the 82586 will utilise a significant bus bandwidth when operating on a net running at 10 Mbps (note: this is not simply the serial data rate divided by the parallel bus width). This bandwidth cannot be provided by the ARM processor over the podule bus and so a dual-port memory system has been implemented.

All communication between the ARM and the LANCE is carried out through command blocks in the dual-port RAM (there are no visible registers in the 82586 LANCE). These command blocks and associated data structures are defined and described in Intel's data sheet.

To issue a command to the LANCE the ARM appends the command to the command block list (CBL) in the dual-port RAM. It then raises the channel attention (CA) signal to the LANCE signalling the presence of the new command. The LANCE responds to CA by reading the command from the CBL and executing as required.

The *LAN Components User's Manual* contains a considerably more detailed and comprehensive description of the operation of the LANCE.

The control register

The control register contains four bits:

Reset (RST) Bit 0.

This bit controls the RESET pin on the LANCE. This bit is set (LANCE reset) on system power-up/hard reset or writing to the control register with this bit logic 1. This bit is cleared (and the LANCE released from the reset state) by writing to the control register with this bit logic 0.

Loop-Back (LB) Bit 1

This bit selects the loop-back mode of 82501 SAI chip. This bit is set and the SIA chip put into loop-back mode by the ARM writing to the control register with this bit logic 1. This bit is cleared (SIA taken out of loop-back mode) on system power-up/hard reset or writing to the control register with this bit logic 0.

Channel Attention (CA) Bit 2

This bit generates a correctly timed CA pulse when the ARM writes to the control register with this bit logic 1. No CA pulse is generated if the ARM writes to the control register with this bit logic 0.

Clear Interrupt (CLI) Bit 3

This bit clears the podule interrupt flag and removes the podule interrupt when the ARM writes to the control register with this bit logic 1. The podule interrupt and flag are unaffected if the ARM writes to the control register with this bit logic 0.

Each bit in the control register is not independent and when writing to a particular bit, the remaining three must be valid. The remaining 12 bits are ignored by the hardware (zero is recommended).

Podule identification PROM

The podule identification PROM contains the following information:

- the Acorn podule identity number (03)
- the interrupt (IRQ) flag bit
- the PCB revision number
- the six byte IEEE globally assigned address block
- a CRC to allow the PROM to be validated.

The contents and operation of the interrupt flag are described in *Interrupts* in *Detailed description* below.

Detailed description

Address map

The Ethernet I expansion card address map (offset relative to slot base) is shown in *Table 1* below. The RAM buffer occupies the upper half of the podule address space. The ID PROM, page register and control register occupy the lower half.

The LANCE

The 82586 LANCE is a 'scatter-gather' DMA controller type device and is designed to interface to 80186 type processors using a HOLD/HOLDA protocol to resolve arbitration for access to shared memory.

The ARM podule bus cannot easily support a HOLD/HOLDA type interface. This is because the ARM is a dynamic device and cannot be stopped for the required time. (This can be longer than 10 µs during the interframe/interpacket spacing time.) The ARM cannot be given priority and HOLDA deasserted because this will result in the net controller failing to meet the timing requirements of the net protocol due to the increased bus latency. For example, this could result in the failure of the net controller to take part in the back-off and retry sequence following a collision on a heavily loaded net.

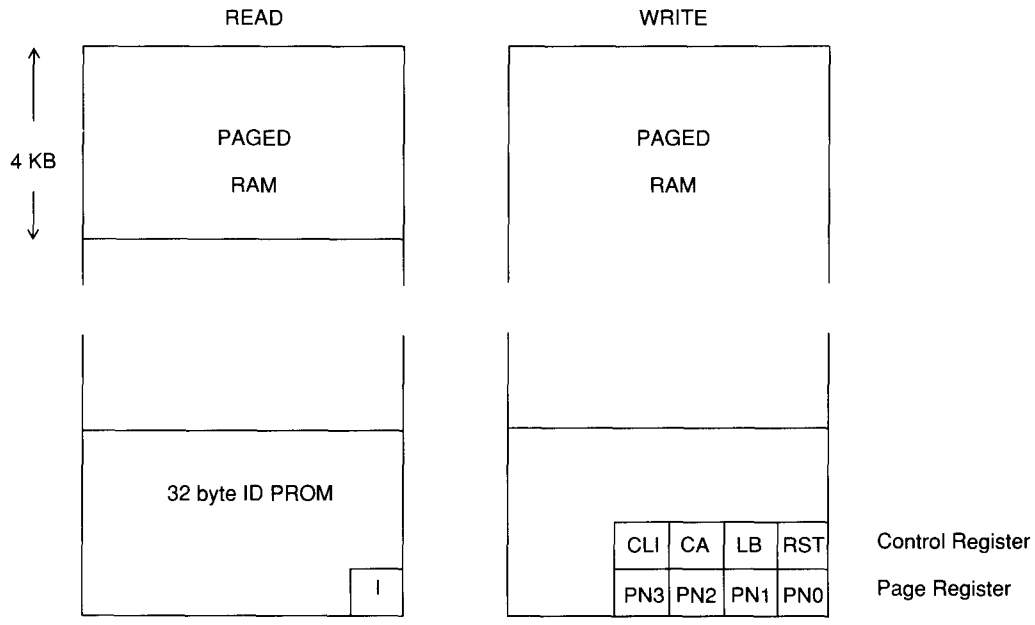
In this design HOLD and HOLDA are wired together and ARM cycles cause wait-states to be inserted into the LANCE bus cycle. This is achieved by removing the READY signal to the LANCE while the ARM is active.

Adopting this scheme avoids the problems outlined above. The ARM is never stopped and the LANCE sees minimal bus latency.

The LANCE ARDY/SRDY input used can be programmed to be either asynchronous/ ARDY and internally synchronised, or synchronous/SRDY and externally synchronised. In this case it is SRDY mode that must be selected. This is achieved by issuing a configure command with the ARDY/SRDY bit set to logic 1. This is important as the LANCE powers-up in ARDY mode.

In certain circumstances the LANCE needs to perform read-modify-write bus cycles with lockout. Using READY to insert wait-states does not allow this. However lockout is only required when the LANCE updates error counts (statistics) and even then a problem only arises when a count overflows and the ARM resets it to zero while the LANCE is in the modify phase of a read-modify-write cycle. This is solved by the ARM reading back the count after it sets it to zero. If the count is still indicting an overflow then a read modify-write cycle was in progress

Table 1: Ethernet I expansion card address map



ID PROM (Base)	= Slot base	(Type 3 access) + 0x0000 (Read only)
Page Register	= Slot base	(Type 2 access) + 0x0000 (Write only)
Control Register	= Slot base	(Type 2 access) + 0x0004 (Write only)
Dual-Port RAM	= Slot base	(Type 2 access) + 0x2000 (Read/write)

and the ARM has to correct the count. Error counts this high indicate a major problem that will require correction so should be a rare event.

The memory bus of the LANCE is operated in 'minimum mode' as the timing parameters for LANCE outputs in this mode are subject to less spread between devices. The pull-up resistors on WR*, RD*, and BHE are required to prevent RAM cycles when the LANCE is inactive.

The LANCE communicates directly with the SIA (IC24) via a serial channel comprising seven signals: TXC, TXD, RXC, RXD, RTS, CRS and CDT. The function of each of these is described in the LANCE data sheet. The Clear-to-Send (CTS*) input is not supported by the SIA and is connected to 0V (enabled).

Dual port RAM

The podule bus provides only a limited space in the address map (8 KB) for each podule. This is insufficient and so a paged scheme has been implemented.

Viewed from the ARM side the RAMs are paged into the top half of podule space by a 'page register'. The four bit page register is split across two PALs (see the section *The PALs* below). Sixteen pages each of 4 KB provide 64 KB in total. This is organised as 32 k x 16 bits (two 32 k x 8 static RAMs). An alternative RAM size of 8 k x 16 bits (two 8 k x 8 static RAMs) can be supported (see the section *Links* below).

The podule address bus (LA2-13) is buffered by two HCT244 (IC66 and IC58) and the podule data bus (BD0-BD15) is buffered by and two HCT245 transceivers (IC15 and IC54). The direction of the data bus transceivers is determined by the podule R/W signal, while both output enables (AAOE and BDOE) are generated by the bus control PAL (IC36).

Viewed from the net controller side, the RAM will be contiguous from location 0x0000 to 0xFFFF. The initialisation root for the controller is 0x0FFFFF6 which is mapped into the RAM at 0xFFF6. The high order address bits are not decoded.

The LANCE address/data bus (AD0-AD15) is demultiplexed by two HCT245 (IC17 and IC22) which use the LANCE ALE signal to latch the address bus. The data bus only requires buffers and two HCT573 transceivers (IC10 and IC32) are used. The direction of the data bus transceivers is determined by the LANCE DT/R signal, while the output enables are generated by the bus control PAL (IC36).

The LANCE is capable of operating on an eight bit bus and is reset to this mode. The LANCE initialisation root (read when released from reset) contains a bit that defines the bus width and this must be set to 0 (=16 bit bus). Until the LANCE reads this it deasserts Byte High Enable (BHE*) and outputs address bits on AD8-AD15 for the entire cycle. To avoid a bus clash BHE* is used to disable the high order data bus transceiver via the bus control PAL (IC36).

Once initialised to a byte wide bus the LANCE only operates on half words (never bytes) so it not necessary to decode the least significant address bit (AD0) to produce separate write strobes for each byte.

Podule identification PROM

The device used is a 32 byte PROM 27LS19 (IC14).

Typical content of an ID PROM is shown in Table 2 overleaf.

The ID PROM shares address and data bus buffers with the RAM. Viewed from the ARM side the ID PROM is byte wide and word aligned.

The podule specification defines two bits in the ID byte to be interrupt flags. This design requires only IRQ interrupts so the FIQ flag is always zero. The IRQ flag is generated by connecting the podule interrupt signal to the most significant address pin. The content of the upper half is similar to the lower half but has the IRQ flag bit set, in this way the interrupt flag is multiplexed 'into' the ID byte.

Bytes 09 - 0E are the six byte Ethernet address unique across all Ethernet equipment from manufacturers worldwide.

The CRC (Bytes 1C - 1F) is calculated on the rest of the PROM (Bytes 00 - 1B) using a 32 bit Autodin - II CRC polynomial. This is the same algorithm as the LANCE uses to perform multicast address filtering (see the section *PROM CRC calculation* below). Since each PROM is unique the CRC is used to perform verification.

The output enable is generated by the bus control PAL (IC36).

The PALs

Three PALs are used in this design:

- the main state PAL (IC29)
- the interrupt and channel attention PAL (IC78)
- the device enable control PAL (IC36).

The main state PAL (IC29)

This PAL implements a state machine which provides timing information for the other two PALS in the design. In addition it produces the two least significant bits of both the page register (PR0 and PR1) and control register (RSTO and LOOP).

The interrupt and channel attention PAL (IC78)

This PAL implements the two most significant bits of both the page register (PR2 and PR3) and control register (CLI and CA).

The device enable control PAL (IC36).

This device decodes the address map to provide various device output enables.

Table 2: Podule identity PROM

	D7	D6	D5	D4	D3	D2	D1	D0	NOTES
1F	C	C	C	C	C	C	C	C	CRC on bytes 00 - 1B
1E	C	C	C	C	C	C	C	C	
1D	C	C	C	C	C	C	C	C	
1C	C	C	C	C	C	C	C	C	
1B	Bytes 11 to 1B = 00								
11									
10	0	0	0	0	0	0	0	1	01 - no FIQs, IRQ = 1
0F	0	0	0	0	0	0	0	0	00 - RSVD
0E	I	I	I	I	I	I	I	I	Unique ID
0D	I	I	I	I	I	I	I	I	
0C	I	I	I	I	I	I	I	I	
0B	1	0	1	0	0	1	0	0	A4
0A	0	0	0	0	0	0	0	0	00
09	0	0	0	0	0	0	0	0	00
08	0	0	0	0	0	0	0	1	01 - PCB rev. eg one
07	0	0	0	0	0	0	0	0	00 - UK
06	0	0	0	0	0	0	0	0	Acorn
05	0	0	0	0	0	0	0	0	
04	0	0	0	0	0	0	0	0	Ethernet
03	0	0	0	0	0	0	1	1	
02	0	0	0	0	0	0	0	0	00 - RSVD
01	0	0	0	0	0	0	0	0	00 - no boot code
00	0	0	0	0	0	0	0	0	00 - no FIQs, IRQ = 0

The state machine and operation

The state machine has four states; IDLE, SA1, SA2, and SA3 and is clocked from state to state on the falling edge of CLK8, the 8 MHz podule bus clock. The figure below is the state diagram.

The idle state

The state machine enters this state on power-up, hard reset (RST* low), or from the SA3 state. In this state the bus buffers on the ARM side of the dual-ported RAM are disabled and those on the LANCE side enabled. Other outputs such as the page and control register bits remain unchanged. The state machine remains in the idle state until the ARM starts an access (podule select - PS active).

The SA1 state

This state is entered from the idle state only. In this state the LANCE READY signal is disabled, forcing the LANCE to insert wait states if it is active on the bus. The RAM write strobe (RAMWE*) is disabled to prevent writes while the LANCE side of the dual-port RAM is disabled and the ARM side enabled. The state machine exits to the SA2 state unless a reset occurs.

The SA2 state

This state is entered from the SA1 state only. In this state the ARM access is performed and the corresponding device enables are active eg, if a RAM write is performed then the RAM write strobe (RAMWE*) is active. Similarly if a RAM or ID read is required than the RAM or IDOE is active. Writes to the page register or control bits are also performed during this state. READY is still inactive. The state machine exits to the SA3 state unless a reset occurs.

The SA3 state

This state is entered from the SA2 state only. The RAM write strobe (RAMWE*) is disabled to prevent writes while the LANCE side of the dual-port RAM is enabled and the ARM side disabled. The state machine exits to the idle state where any LANCE access that was in progress is completed.

Podule bus cycles

The podule specification requires all ID PROM access to be made using type 3 (sync) IOC bus cycles. All other accesses to the Ethernet podule must be made using type 2 (fast) IOC cycles.

Figure 3 overleaf illustrates a read/write to RAM while the net controller is active. The cycle starts with podule select (PS) active and puts the state machine into the SA1 state on the next clock edge. A description of each state that follows is given above.

It should be noted that *Ready* is always deasserted for three cycles, even if the LANCE is idle. A podule bus access can 'collide' with a LANCE access in five different ways, depending on what state the LANCE is in when the podule bus access starts. These are: PS* while the lance is in states T1 to T4 or idle. The actual number of wait states that the LANCE will insert depends on which of these cases apply. Figures 4, 5, 6 and 7 illustrate the possible cases.

Figure 2: State diagram

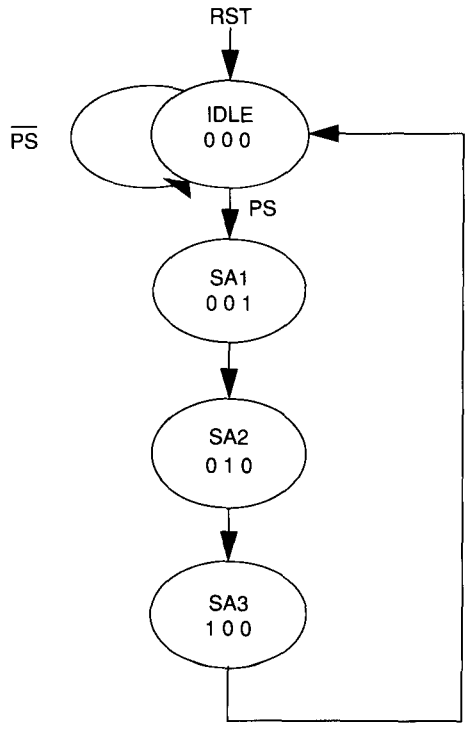


Figure 3: Typical podule bus cycle

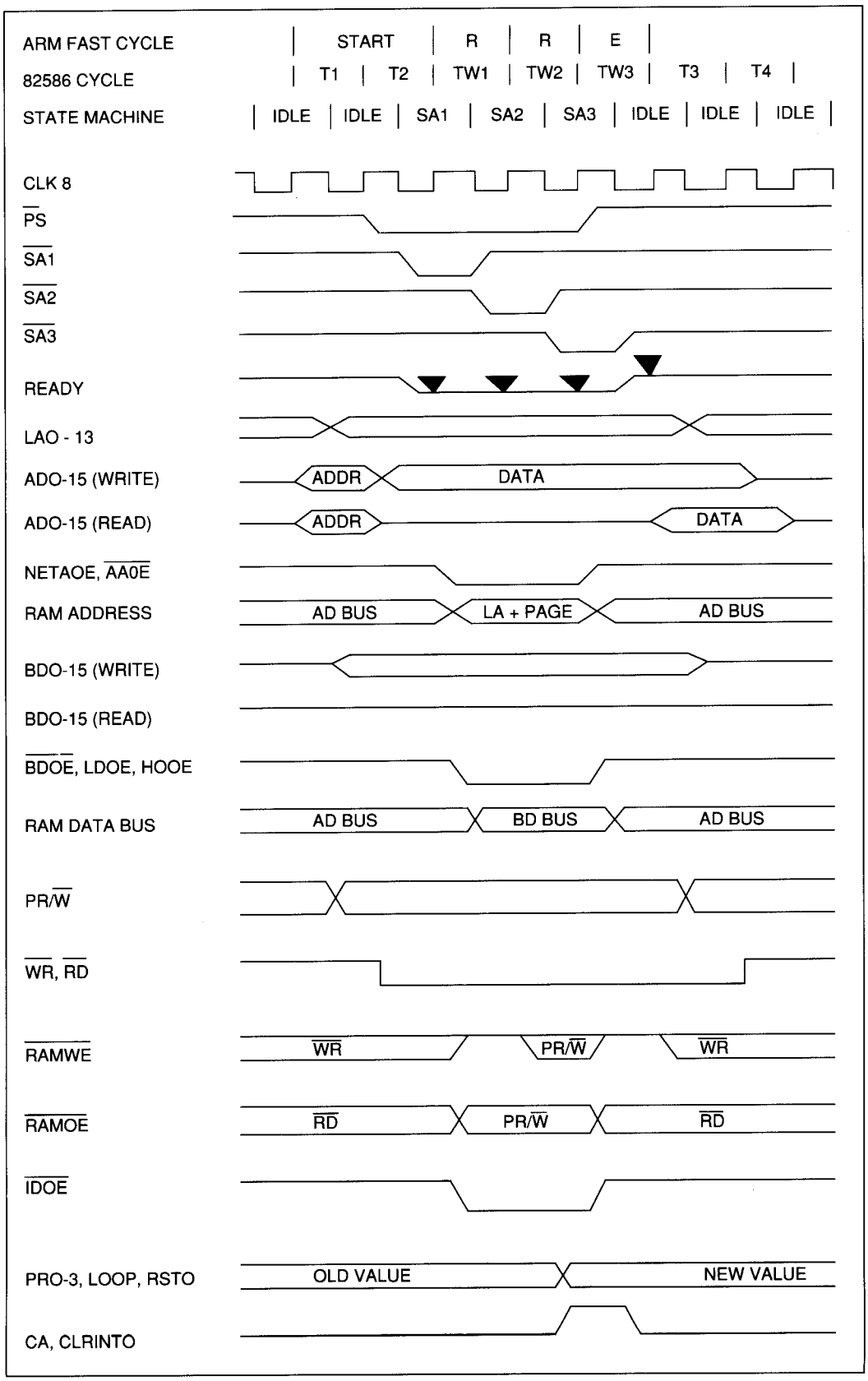
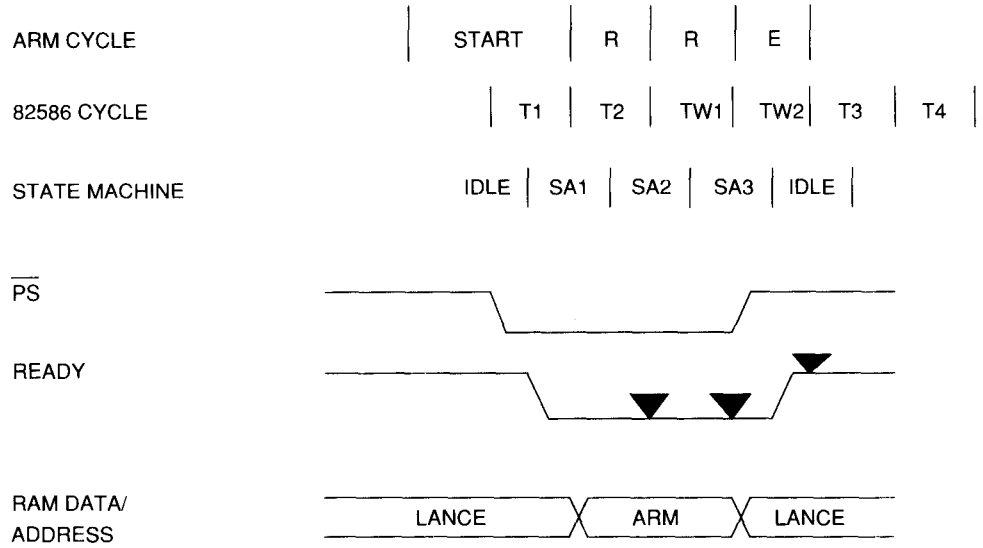


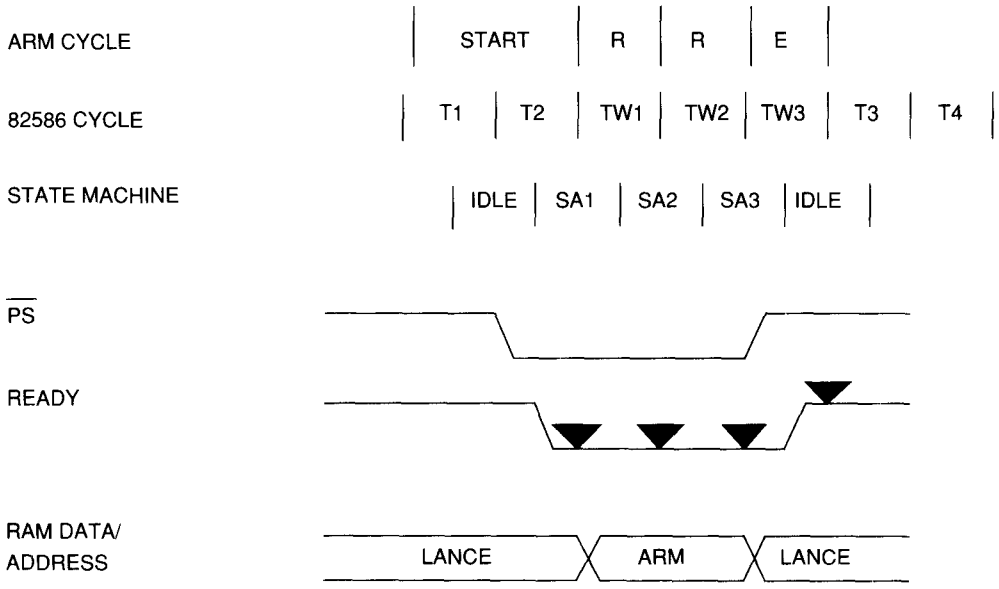
Figure 4: Access collision cases PS* while LANCE is in T1



PS* while the LANCE is in T1

The LANCE samples READY deasserted at the end of T2 (SA2), and then again at the end of TW1 (SA3), so in this case two wait states are inserted.

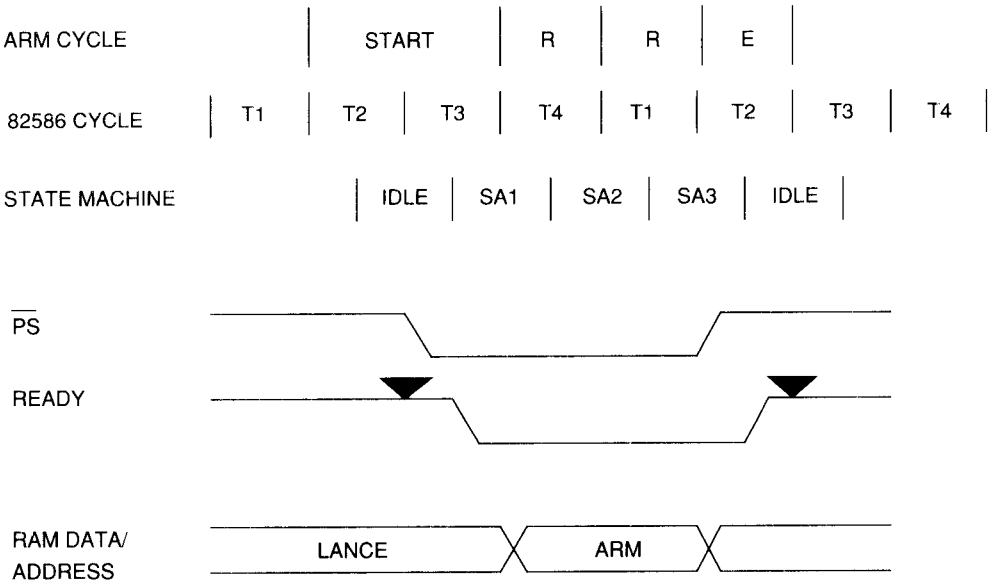
Figure 5: Access collision cases PS* while LANCE is in T2



PS* while the LANCE is in T.2

The LANCE samples READY deasserted at the end of T2 (SA1), TW1 (SA2), TW2 (SA3), so the maximum of three wait states are inserted.

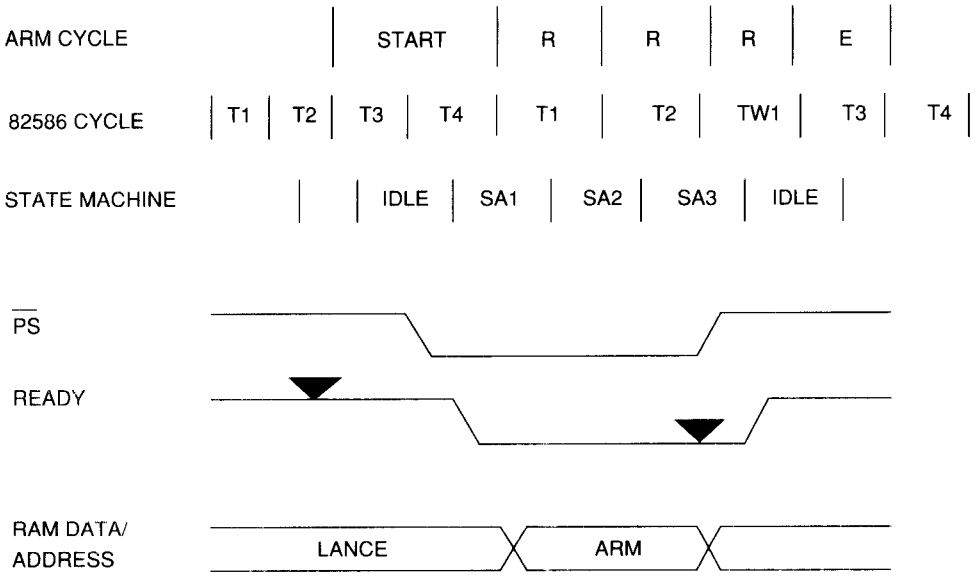
Figure 6: Access collision cases PS* while LANCE is in T3



PS* while the LANCE is in T3

In this case READY is still active when the LANCE samples it at the end of T3 (idle). This is the last time that the LANCE does this for the current cycle so the LANCE cycle completes before the podule bus cycle starts. Note that the LANCE is not active on the RAM bus during T4.

Figure 7: Access collision cases PS* while LANCE is in T4



PS* while the LANCE is in T4

Since the LANCE does not require the bus during T4 no further wait states are inserted in the current cycle. However T1 of the next cycle could follow T4 and one wait state will be inserted into this LANCE access.

PS* while the LANCE is idle

If the LANCE remains idle while the podule bus cycle occurs then there is no collision and the LANCE ignores the READY signal. This case is not illustrated.

A read from the podule ID PROM or write to the control or page register is similar to a RAM cycle. To simplify the bus design the LANCE is removed from the RAM buses during cycles to these devices.

Bus design note

The cycle stealing scheme should guarantee that the LANCE never has insufficient bus bandwidth or sees excessive bus latency to the extent that it cannot service the net or fails to meet the IEEE timings. Even when the ARM continuously accesses the RAM. The following gives the reasoning behind this statement:

Assumptions:

Net Clock	=	10 MHz
Bus Clock	=	8 MHz
LANCE FIFO size	=	16 bytes

HOLDA is wired to HOLD so:

Bus Latency	=	0 cycles
IEEE Interframe Space Time	=	9.6 µS

Criteria:

1 FIFO must not over/underrun.

FIFO fill/empty time from serial side:

= 8 (bits) * 16 (bytes) * 100E-9 (bit time)

= 12.8 µs

FIFO empty/fill time from parallel side:

= 8 (Word transfers)

* (4 (standard 8 MHz cycles) + Nwait (wait cycles))

* 125E-9

= 4 µs (if Nwait = 0)

= 7 µs (if Nwait = 3)

= 8 µs (if Nwait = 4)

2 The LANCE must be in a position to transmit by the end of the interframe spacing time.

With a Fp/Fs ratio of 8 MHz/10 MHz (0.8):

16*Nwait + Nlatency must be less than or equal to 80.

If HOLDA = HOLD then Nlatency = 0

and

Nwait <= 5

So this strategy works if we can keep the number of wait states (Nwait) less than or equal to five per access. In the current design three are used and this is unlikely to change.

Interrupts

The podule interrupt (PIRQ) is level triggered. However, the interrupt signal (INT) from the LANCE is designed for use with edge triggered interrupt controllers. If the net controller detects a second interrupting condition just after the first is raised, it will drop and reassert INT. The situation could arise where the podule manager (software) may scan the slots and find no IRQ flag set.

The above problem is prevented by latching INT in the interrupt and channel attention PAL (IC78) and using the latched signal INTO to generate the flag. The clear interrupt (CLI) bit in the control register is used to clear the latch.

Latching INT introduces another problem, which is eliminated by a feature of the 82586 LANCE. If a second interrupt occurs after the processor has read the status word in the SCB, but before the first is cleared, then the second interrupt would be missed. However, if the interrupt is cleared at the same time as the channel attention (signalling the acknowledge command) is issued, the LANCE will respond by deasserting INT and reasserting if the second interrupt was not acknowledged because it was missed. It is recommended to set CA whenever CLI is set.

Figure 8: Example interrupt cycles

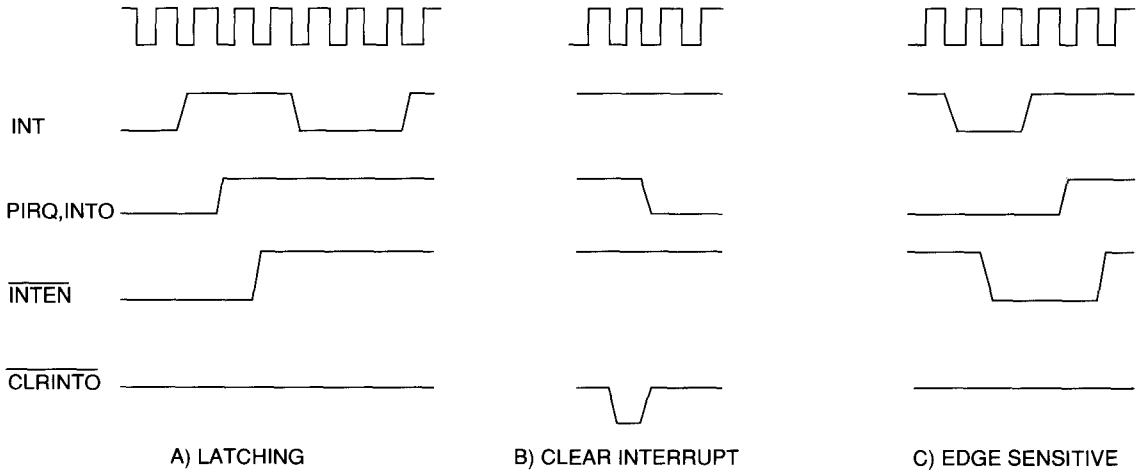


Figure 9: State diagram for INTO/PIRQ*

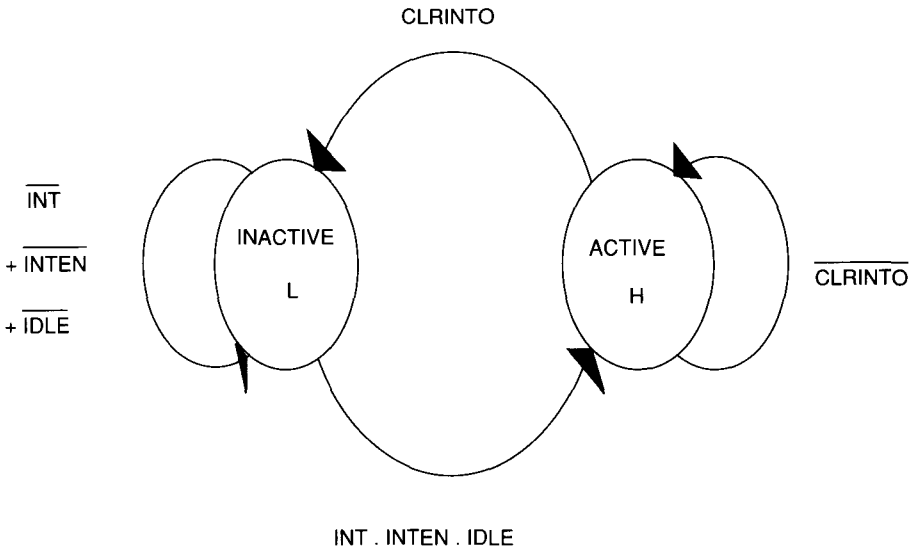
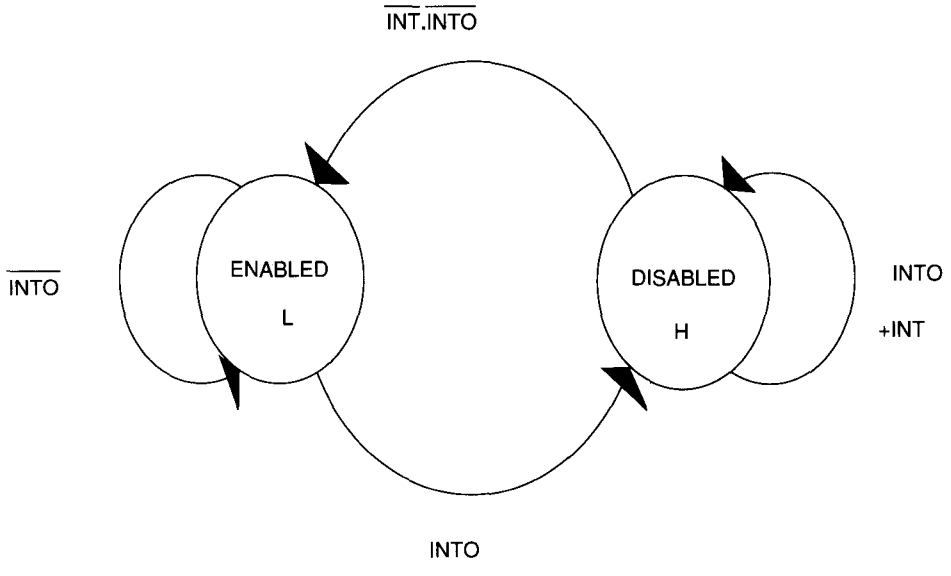


Figure 10: State diagram for INTEN*



Links

The Ethernet I PCB should be viewed from the component side with the 96 way podule bus connector on the left and the rear panel on the right. When viewed like this, west is to the left, east the right, north the top and south the bottom.

LK1 and LK2 select the RAM size

If 32 KB devices are fitted (normally) the links should both be south. 8 KB devices will not normally be fitted but in this case LK1 and LK2 should be north.

LK3 to LK8 select Ethernet or Cheapernet.

For Ethernet operation the links should be west (link pin a to pin b). For Cheapernet operation the links should be east (link pin b to pin c).

LK9 is tracked south and not fitted on production units.

See data sheets for the 82502 for use.

PROM CRC calculation

The following is a code fragment in the C programming language that calculates and validates the Ethernet PROM checksum.

```
/* To calculate and check the PROM checksum */
int ROM_chk(vector)                                          /* array 0..32 bytes*/
u_char vector[32];
{  register int i,j;
  register unsigned chk = -1;                                /* Set the CRC register*/
                                                             /* to FFFFFFFF*/
  register unsigned byte;                                    /* temp
 for (i = 0; i < 28; i++) {                                  /* CRC on bytes 0..28*/
     byte = vector[i];
     for (j = 0; j < 8; j++) {
        if (((byte & 1) ^ (chk >> 31)) != 0)                 /* IF feedback = 1*/
           chk = (chk << 1) ^ (0x04C11DB7);                  /*    shift and EOR taps*/
        else                                                 /* ELSE
           chk = (chk << 1);                                 /*    just shift*/
        byte = byte >> 1;                                    /* next bit*/
     }
  }

/* chk is now the calculated CRC */
/* Now get CRC from PROM */
 byte = (vector[31] << 24) | (vector[30] << 16) | (vector[29] << 8) |
         (vector[28] << 0);
/* Test to see if the same */
 if (byte != chk) return (FALSE); /* checksum error*/
  else return (TRUE);
}
```

Ethernet II expansion card

The IEEE 803.2 standard supports two different versions for the media:

- 10BASE5 (commonly known as Ethernet)
- 10BASE2 (thin-wire Ethernet, or 'Cheapernet').

These can be used separately, or together in a hybrid form. Both versions have similar electrical specifications and can be implemented using the same transceiver chip. Thin-wire Ethernet is the lower cost version and is user-installable. Main differences are in the segment length, network span and nodes per segment, with thin-wire Ethernet having only one-third of the performance. The capacitance per node and the cable cost are however much less.

The Ethernet expansion card has been designed to provide the physical and media access control layer functions of the local area network as specified in IEEE 802.3 standard. This standard is based on the access method known as Carrier-Sense Multiple Access with Collision Detection (CSMA/CD). In this scheme, if a network station wants to transmit, it first 'listens' to the medium; if someone else is transmitting, the station defers until the medium is clear before it begins to transmit. However, two or more stations could still begin transmitting at the same time and give rise to a collision. When this happens, the two nodes detect this condition and back off for a random amount of time before making another attempt.

System considerations

Bus Latency is the maximum time between the NIC (Network Interface Controller) assertion of BREQ and the system granting BACK. This is of importance because of the finite size of the NIC's FIFO. If the bus latency becomes too great, the FIFO overflows during reception, and becomes empty during transmission. The Bus Utilization is a fraction of the time the NIC is the master of the Ethernet podule internal bus, and this should be minimised. The lowest bus utilization occurs when the bursts of data across the podule interface are as long as possible. This requires the threshold as high as possible, and Empty/Fill mode is used. The determination of the threshold is related to the maximum bus latency the system can guarantee.

A DMA set up and recovery time is associated with each burst, hence when longer bursts are used, less bus bandwidth is required to complete the same packet.

Hardware overview

The Ethernet II expansion card has been designed around the National Semiconductor Chip Set. This provides all the functions necessary to implement an IEEE 802.3 (Ethernet/thin-wire Ethernet) interface on a host computer or a peripheral device. As there is no direct DMA memory path across the podule bus, data is transferred via a static RAM local buffer. Since both the ARM and the DMAC will have access to the Ethernet II expansion card internal bus, some arbitration is required.

Dual-port memory equivalent

This configuration makes use of the NIC's remote DMA capabilities, and requires only a local buffer memory and a bi-directional I/O port. The high priority network bandwidth is decoupled from the system bus, and the system interacts with the local buffer memory using a lower-priority bi-directional I/O port. When a packet is received, the local DMA channel transfers it into the buffer memory, part of which has been configured as the receive buffer ring. The remote DMA channel transfers the packet on a byte by byte basis to the I/O port. At this point the data is transferred through an asynchronous protocol into main memory.

Remote DMA

The remote DMA channels work in both directions; pending transmission packets are transferred into the local buffer memory, and received packets are transferred out of the local buffer memory. Transfers into the network memory are known as remote write operations, and transfers out of the local buffer memory are known as remote read operations. A special remote read operation, Send Packet, automatically removes the next packet from the receive buffer ring. Both the starting address and the length are set before initiating the remote DMA operation. The remote DMA operation begins by setting the appropriate bits in the Command Register. When the remote DMA operation is complete, the RDC bit in the Interrupt Status Register (ISR) is set and the processor receives an interrupt. When the Send Packet command is used, the controller automatically loads the starting address and byte count from the receive buffer ring for the remote read operation. Upon completion it updates the boundary pointer for the receive buffer ring. Only one remote DMA operation can be active at any time.

Hardware components

The Ethernet II expansion card can be divided into five major blocks – see the circuit block diagram below. The five major blocks are as follows:

1 Decode and cycle access control:

Carrying out address and register decoding, control of the local buffer (latched or transparent mode) and all the required read/write signals. The type of access cycle required may be extended if bus arbitration is needed.

2 Podule and Ethernet identification:

A PROM containing the ID of the type of podule (expansion card) that is fitted, with the address of the

interrupt location, the Ethernet ID of the particular board (each PROM is programmed with a different number) and required driver code to run under RISC OS. It is page addressed by writing to 'mode' latch. System reset sets to page zero.

3 Data Buffer:

Static RAM memory. Memory access is completely controlled by the NIC controller which performs the memory management. Data is transferred between the controller and SRAM using local DMA, and between the SRAM and the PORT by remote DMA.

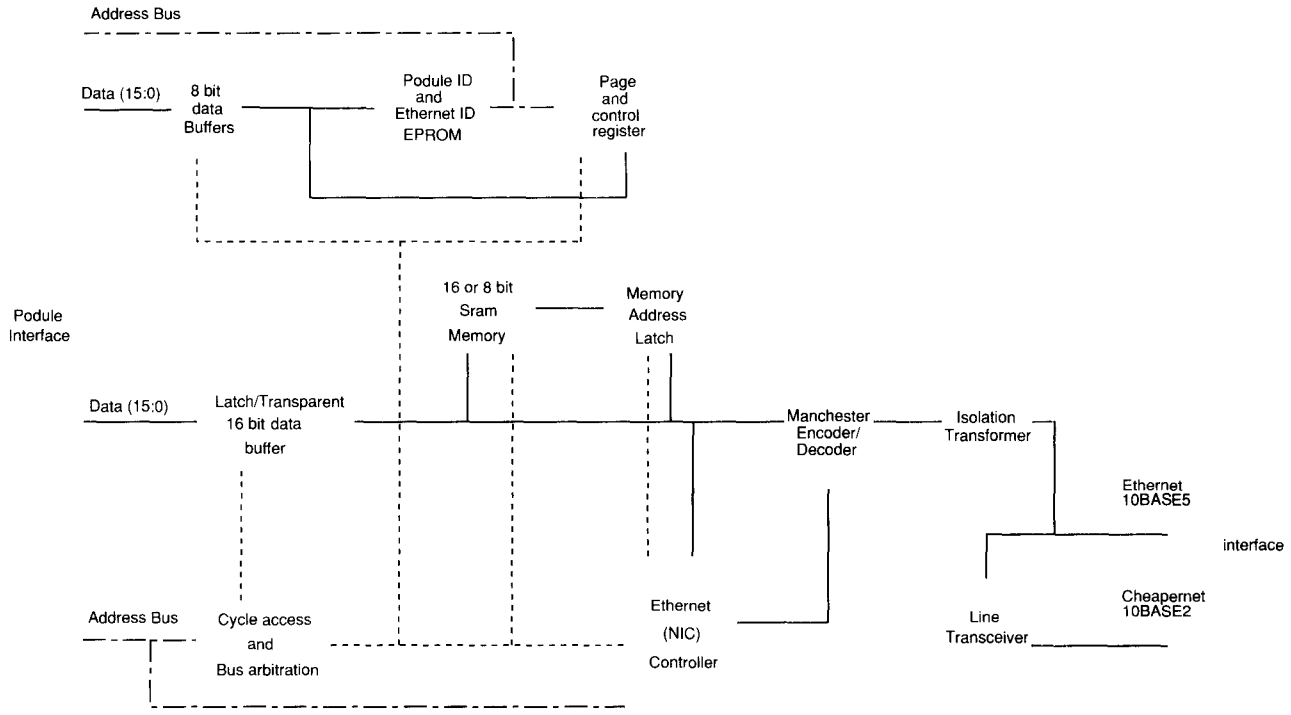
4 NIC controller:

Provides the required data rate with a minimum of control overhead. This is a key element in the design. Once set up it performs many of the Ethernet functions without requiring processor help, only producing an interrupt when a packet has been completely received or transmitted.

5 IEEE802.3 Interface Components:

Providing the Manchester encoding/decoding, high voltage isolation and line drivers for the thin-wire Ethernet interface.

Ethernet II block diagram



Circuit component details

Decode and cycle access control

The Ethernet II expansion card has hardware in both podule space and Module space. The podule section consists of the ID/RISC OS driver EPROM, the interrupt status register and the EPROM page register. The podule hardware is kept isolated from the Module hardware so that accesses to the Interrupt Status Register and Page Register do not affect any DMA transfers in progress on the Ethernet podule internal bus.

The podule memory map is shown below:

Address P0	LA13	LA12	use
03343000	1	1	WRITE Page Register READ not defined
03342000	1	0	WRITE not defined READ Interrupt status
03341000	0	1	
03340000	0	0	EPROM

The Interrupt Status register is as follows:

bit7	=	X	Not used
bit6	=	X	Not used
bit5	=	X	Not used
bit4	=	X	Not used
bit3	=	X	Not used
bit2	=	X	Not used
bit1	=	X	Not used
bit0	=		interrupt pending

When the Ethernet II expansion card generates an interrupt, the 'podule manager' will interrogate the status register (as defined by the podule ID) to check for bit 0 set active low.

In Module space the ARM has access to the Ethernet controller and the data transfer I/O Port. When a local DMA transfer between NIC and SRAM is in progress, the ARM may still access the NIC or I/O Port in the normal manner, simply by reading and writing to them. All arbitration required to gain access to the Ethernet II expansion card internal bus (when accessing the NIC) or waiting for data to be ready at the port, is carried out transparently by stretching the MEMC cycle.

The NIC has 46 registers (normally accessed using address bits RA0 through RA3 of the host processors data bus. RA0 through RA3 on the NIC are connected to LA2 through LA5) which provide the flexibility and programmability to handle both the Ethernet interface and also the interface to the local memory and controlling processor.

The I/O Port is used to transfer packets of data to and from the Ethernet/thin-wire Ethernet via the podule interface, by simply writing or reading the required data file length in 16 bit wide words. The individual bytes being transferred automatically between the Port and Network via the NIC and SRAM.

The Module memory map address is shown below:

Address P0	LA13	LA12	use
03003000	1	1	NIC controller using LA5–LA2
03002000	1	0	Data transfer I/O Port
03001000	0		
03000000	0	0	

Podule and Ethernet identification

The ID/RISC OS driver PROM has been laid out to give from 8kB to 512kB of code space. The host cannot directly address the full PROM and therefore is operated in a page mode by writing the required page to the page register. The page register is set for page zero by power on reset. The top two bits of the page register being used for 'Lr_w' (access to the I/O Port is set for reading or writing a packet) and 'Srst' (software internal reset). The page register is not cleared by the 'software internal reset'.

The page register is as follows:

bit15	=	Srst (active low)
bit14	=	Lr_w (active high – read)
bit13	=	
bit12	=	
bit11	=	
bit10	=	
bit9	=	
bit8	=	EPROM page address bit 8
bit7	=	EPROM page address bit7
bit6	=	EPROM page address bit 6
bit5	=	EPROM page address bit 5
bit4	=	EPROM page address bit 4
bit3	=	EPROM page address bit 3
bit2	=	EPROM page address bit 2
bit1	=	EPROM page address bit 1
bit0	=	EPROM page address bit 0

Local Buffer Memory

The buffer memory consists of two 8k x 8 (up to 512k x 8 for SRAM source flexibility) static RAMs which give a 16 bit data transfer across the podule interface, hence maximizing the podule bandwidth. The data buffer is completely controlled by the NIC controller, which performs all the memory management in a ring buffer format. Pointers to the memory are updated as required (but can be accessed via the NIC registers if necessary). The data buffer is transparent as far as data transfers across the podule interface are concerned.

NIC Controller

The National Semiconductor Network Interface Controller provides all the functions necessary to implement all Media Access Control (MAC) layer functions for transmission and reception of packets in accordance with the IEEE 802.3 standard. All bus arbitration and memory support logic and two DMA channels are integrated into the NIC. The local DMA channel transfers data between the internal controller FIFO and local memory. On transmission, the packet is transferred from local memory to the FIFO in bursts. Should a collision occur, the packet is re-transmitted with no processor intervention. On reception, packets are transferred from the FIFO to the receive buffer ring. A remote DMA channel is provided to transfer between local buffer memory and system memory. Full details for operating the NIC are contained in the data book (see the *Bibliography* at the end of this section).

IEEE802.3 Interface Components

These are the components concerned with the Ethernet/Thin-wire Ethernet interface. They include the 20MHz oscillator (providing the required transmit and receive clock), the Manchester encoder/decoder, DP8391 (to produce the required signals), the transceiver/line drivers,DP8392 (required to provide thin-wire Ethernet signals) and components to provide isolation such as the DC to DC convertor, line transformers, termination resistors, capacitors and a diode as required.

PALs

There are four PALs used:

- Decode
- Intbuf
- Memcpal
- Natfix.

Decode (0273,271)

As its name suggests, this PAL decodes podule and module addresses to produce chip select signals. It enables reading of the EPROM, writing to the page register, reading interrupt status, and read/write operations to the NIC controller main podule interface functions. It also defines whether podule or DMAC have control of the bus. The PAL's function is shown by the state flow diagram below.

Intbuf (0273,272)

The 'intbuf' PAL, in conjunction with the 'memcpal' PAL, form the core to the Ethernet podule bus arbitration logic. Intbuf produces the interrupt control and all the functions required to control the I/O Port (HCT646s, which are used in both latched and transparent mode, depending on the type of access active).

Memcpal (0273,273)

The 'memcpal' PAL, working in conjunction with the 'intbuf' PAL, produces all the podule interface (MEMC) required read and write pulses. The Ethernet controller has two main modes of operation – Bus Master (while performing DMA) and Bus Slave (while its internal registers are being accessed. These two modes require two different types of access cycle (a different bus arbitration is used). Within these two modes a read or a write cycle may be in operation. The PAL's function is shown by the state diagram overleaf

The internal reset will set this PAL to the 'Idle' state. It remains in this state until a MEMC cycle is decoded. From the 'Idle' state it may enter one of four states:

- Slave Read
- Slave Write
- Master Read
- Master Write.

On entry to one of these states, a complete cycle will follow. Whichever state it has entered, it will remain in that state while the bus arbitration function is completed. Once access has been granted, the cycle continues, producing read or write pulses and MEMC signals (including waiting during interrupts) as required.

Natfix (0273,274)

The National Semiconductor NIC Ethernet controller requires care to be taken when trying to access its internal registers via the control signal Chip Select. The PAL Natfix is used to monitor the controller's use of the bus and then hold back any access to the registers while the controller is using the bus. It similarly holds back the controller during a register access, and has the effect of making sure that Chip Select doesn't become active on a rising edge of the 20MHz clock.

Summary

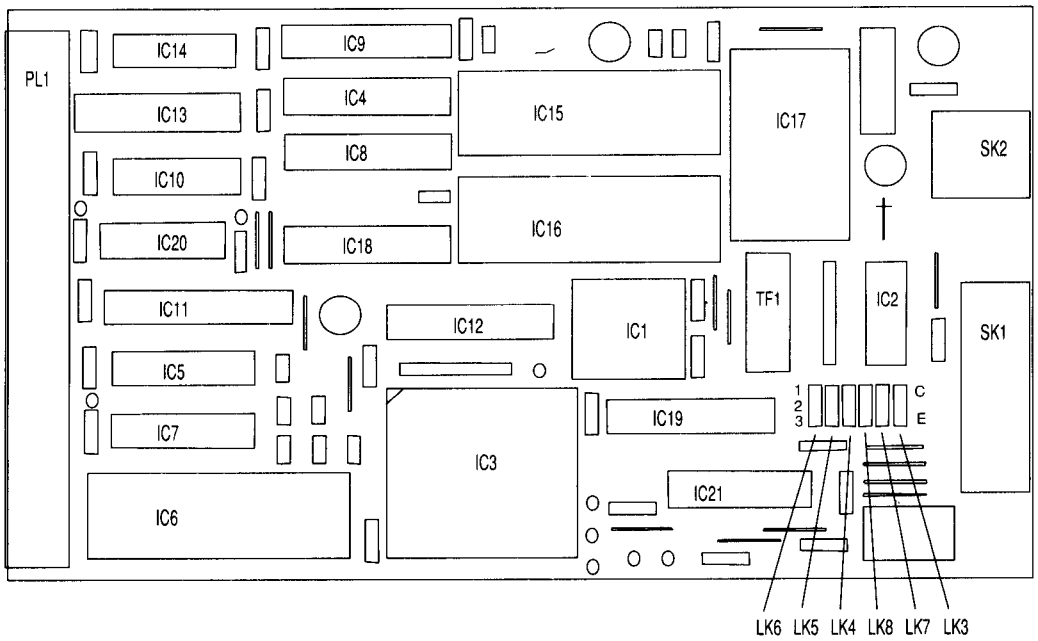
The Ethernet II expansion card hardware design tries to be as transparent, in terms of data transfer, as possible. Where design requirements have allowed, flexibility has been given to the way the software can use this hardware platform, at the same time trying to maintain minimum system overhead. Much of the flexibility of the design is achieved by the use of the DP8390 (NIC), which is a complicated device containing several internal registers allowing software to dictate operation. Therefore access to the Ethernet/Cheapernet LAN is achieved by software drivers that firstly prime the device by direct access and then leave the expansion card to run free, requiring only burst data transfers across the podule interface, an interrupt being used when intervention is required.

Bibliography

The following publications will be of interest to technicians and users wanting to find out more about Ethernet and the Acorn Ethernet II card:

- ANSI/IEEE Std 802.3 – 1985 ISO draft International Standard 8802/3 ISBN 0-471-82749-5, Carrier Sense Multiple Access with Collision Detection (CSMA/CD) access method and physical layer specifications.
- National Semiconductor data sheet for the DP8390C NIC. In the National Semiconductor Data Communications, Local Area Networks and UARTs Advanced Peripherals Handbook, available from National Semiconductor (UK) Ltd, 301 Harpur Centre, Horne Lane, Bedford MK40 1TR.
- 'A-series Podules', specification issue 2.0, available as an application note from Acorn Computers Limited (address at the front of this manual).

Ethernet/Cheapernet links



SCSI interface

Overview

Workstation models which provide a SCSI interface do so by means of a SCSI expansion card (*Podule*) plugged into an expansion slot in the backplane.

The Small Computer System Interface (SCSI) is a high-speed interface for use mainly with mass storage devices such as hard discs, tape drives or CDs. It is an asynchronous bus capable of data transfer rates up to 4MB per second. The bus cable is a 50 way cable consisting of:

- 30 Ground wires
- 9 Control signals
- 8 Data signals
- 1 Data parity signal
- 1 Terminator power line
- 1 Not Connected.

The pin allocation for the two standard SCSI bus connectors can been seen on the schematic.

Communication on the bus is between two devices, an initiator and a target. In the most common case the initiator will be the host computer and the target will be a hard disc drive. The first task for the initiator is to select the required target. There can be up to eight devices on a single SCSI bus, each having its own unique select code.

This select code is simply a different single bit of the data bus allocated to each device on the SCSI bus. Generally the host computer uses select code seven (data bit 7) and the first target will use select code zero (data bit 0).

Having selected its target, the host then transfers a small group of bytes known as the Command Descriptor Block (CDB) to the target. The CDB defines the action to be taken by the target. In the case of a disc drive this will usually be to send some data to the host, or to receive some data from the host and write it onto disc.

The SCSI bus will go through many 'Phases' during the execution of such a command and the target may even release the bus (or 'Disconnect') during the execution of a command (for example if a 'Seek' is required by a hard disc drive), thus allowing the host (or initiator) to initiate commands on other targets.

The complexities of SCSI Bus Phases, handling target disconnections etc, can be all taken care of by single chip SCSI controllers.

For further details of the functions of the SCSI bus, refer to the ANSI Standard X3.131-1986 and the data sheet for the SCSI controller used in the Acorn SCSI expansion card (see the Bibliography).

Circuit description of the SCSI expansion card (Issue 2+)

Maximum performance is achieved from the Acorn expansion bus by the use of the STM (STore Multiple) and LDM (LoaD Multiple) ARM assembler instructions to transfer data to and from a peripheral device. These instructions, coupled with the full use of the 16 bit I/O data bus, will provide a maximum data transfer rate of 8 MB per second. Unfortunately these commands cannot be used to transfer data to and from the SCSI controller chip directly, because it cannot be predicted whether or not the WD33C93A (the device used in this design) can accept or provide the mandatory number of bytes for the relevant instruction.

Furthermore, the WD33C93A is an 8 bit device, hence some kind of funnel hardware is required to couple the 8 bit bus to the 16 bit I/O bus.

The solution to these problems is to have buffer memory on the expansion card, accessible by both the WD33C93A and the ARM processor. This dual porting of the buffer memory is the most complex aspect of the circuit and is therefore dealt with separately.

The main elements of the SCSI Expansion card are:-

- Western Digital WD33C93A SCSI Bus Controller
- NEC 71071 DMA Controller
- two 32K by 8 bit Static RAMs
- EPROM containing the driver software
- four PAL devices controlling ARM access to the card
- SCSI bus connector, termination resistor packs, and filters
- data and address bus buffers and latches.

The SCSI Expansion Card (SEC) has hardware in both Podule (expansion card) I/O space and Module I/O space. The podule section consists of the ID/RISC_OS driver EPROM, the interrupt status register and the EPROM page register. This page register is also used for the SRAM (Static RAM buffer memory) page. The podule hardware is kept isolated from the Module hardware to allow accesses to the Interrupt Status Register (ISR) and the Memory Page Register (MPR) not to interfere with any DMA process that may be taking place between the SCSI Bus Interface Controller (SBIC) and the SRAM.

The EPROM circuit permits from 8 KB up to 128 KB of code space, the top two bits of the MPR being used for interrupt enable (IE) and user reset (UR). The IE is 0 by default and has to be set to 1 before any interrupts can be generated by the SEC. The UR bit is also 0 by default and if set to 1 will cause the internal reset line (IRST) in the SEC to become active. The DMAC has a minimum reset period of 2tCYK (250ns) and the SBIC has a minimum reset period of 1µs. The MPR is not cleared by the IRST signal. A link option does allow the SCSI bus reset to control the IRST, should the card be required to act as a target. The SBIC will inform the host processor that a reset has occurred.

The final section of the podule hardware is the interrupt control logic. There are two sources of interrupts within the SEC, the DMAC and the SBIC. The DMAC will issue a terminal count (TC) pulse at the end of a data transfer which will be latched by the ISR, and may be subsequently read at any time by the ARM processor. SBIC interrupts are latched within the SBIC, but can be monitored in the ISR. DMAC interrupts remain latched until the Clear Interrupt (CLRINT) address is written to. SBIC interrupts remain latched until appropriate action is taken by the host. The two interrupt sources are combined in a PAL to form a common PIRQ.

The address map for podule slot 0 fast access is given below:

Address P0	LA13	LA12	Use
03343000	1	1	Write MPR, UR, IE
03342000	1	0	Read ISR Write CLRINT
03341000	0	1	EPROM (Read Only)
03340000	0	0	

Bit	MPR Bit Allocation	ISR Bit Allocation
7	1 = User Reset	X Not used
6	1 = Interrupts Enabled	X Not used
5	EPROM Page Address	X Not used
4	EPROM/SRAM Page Address	X Not used
3	EPROM/SRAM Page Address	1 = SBIC Interrupt
2	EPROM/SRAM Page Address	X Not Used
1	EPROM/SRAM Page Address	1 = DMAC Terminal Count Interrupt
0	EPROM/SRAM Page Address	1 = SEC Requesting IRQ

In Module address space the ARM has access to the SRAM via a 16 bit data bus and addresses it as 16 4KB pages (8K addresses 16 bits wide, every 4th address), using the MPR located in podule address space.

LA2 of the podule bus is connected to A0 of the SRAM, so that the lower 16 bits of the ARM registers will be stored in consecutive addresses when an STM instruction is used.

The DMAC and the SBIC are also memory mapped but only have 8 bit data buses. The DMAC has many registers which are normally accessed using address bits A0 through A7 of the host processor address bus. Due to the funnelling required to exchange data between 8 bit and 16 bit data buses, the DMAC addressing has had to be mapped rather unusually. A1 through A7 on the DMAC are connected to LA2 through LA8, and A0 on the DMAC is connected to LA9.

Thus the mapping becomes:

Normal Offset	SEC Offset	Register
0000	0000	Initialise
0001	0200	Select Channel to Program
0002	0004	Transfer Count Low
0003	0204	Transfer Count High
0004	0008	Transfer Address Low
0005	0208	Transfer Address Mid
0006	000C	Transfer Address High
0007		Unused
0008	0010	Device Control Register 1
0009	0210	Device Control Register 2
000A	0014	Mode Control Register
000B	0214	Status Register
000C	0018	Temporary Register Low
000D	0218	Temporary Register High
000E	001C	Request Register
000F	021C	Mask Register

The SBIC is used in the indirect addressing mode where LA2 is used as A0 to select between control registers and the address register (see data sheet).

When a DMA transfer between SBIC and SRAM is in progress, the ARM may still access the DMAC, SBIC or SRAM in the normal manner simply by reading or writing to the appropriate address. All arbitration required to gain access to the SEC internal buses is carried out transparently by stretching the MEMC I/O cycle (see the podule bus specification). In the case of an STM and LDM instruction only the first access is stretched to gain control of the SEC buses. The ARM will normally retain control of the SEC buses during video DMA interruptions.

Module Address Map:

Address P0	LA13	LA12	Use
03003000	1	1	DMAC
03002000	1	0	SBIC
03001000	0	1	SRAM
03000000	0	0	

SRAM paging is exactly the same as EPROM paging.

Component identification on the SEC

EPROM address lines from the ARM podule bus are unbuffered. This allows them to operate during DMA. The extra address lines are provided by the MPR (IC17

HCT273), which are directly connected to the EPROM, but isolated from the SRAMs by IC3 (HCT541). The EPROM (IC5) and the ISR (IC15, PAL 0273,215) also have an 8 bit data bus buffer (IC2 HC245) separate from that used for the SRAMs, DMAC and SBIC. Again, this allows ARM access independent of DMA activity. IC4 (HC245) and IC6 (HC245) provide 16 bits of data bus buffering for the SRAMs (IC13 and IC11), as well as the DMAC and SBIC. IC7 (HCT573) is used to hold the upper 8 address bits for the DMAC during DMA transfers, and IC8 (HC245) routes the data to the correct SRAM, depending on the state of A0. The DMAC 'sees' the SRAM as 64K by 8 bits, whereas the ARM 'sees' the SRAM as 32K by 16 bits. IC17 (uPD71071) is the DMAC and IC16 (WD33C93A) is the SBIC.

All address decoding is taken care of by IC9 (PAL 0273,216). The task of arbitration for access to the SRAM is shared by IC15 (PAL 0273,213) and IC14 (PAL 0273,217), IC18 (PAL 0273,219), and IC12 (PAL 0273,218). IC15 [IC14] also generates the IOGT and BL signals required by the podule Bus, while IC12 handles the I/O and memory read and write lines (IORD, IOWR, MEMR, MEMW). There are various link options on the SEC and they are listed below:

Issue 2+ expansion card:

EPROM size select						
EPROM	LK1	LK2	LK3	LK4	LK5	LK7
27128	C	O	O	O	C	C
27256	O	O	O	C	C	C
27512	O	C	O	C	O	C
27C101	O	C	C	C	O	O

O - Open C - Closed

Factory fitted links set the size of the Issue 2+ PCB to 27256.

LK6 and LK7 allow the DMAC to perform memory-to-memory transfers.

LK8 and LK9 allow for larger SRAM devices, but these could not be fully addressed by the ARM processor.

LK10 and LK11 switch the reset line for initiator or target mode:

Mode	LK10	LK11
Initiator	O	C
target	C	O

The PCB is factory configured for initiator mode.

The SCSI bus signals are connected from the SCSI bus connector to the SBIC, via filter capacitors clearly visible on the circuit board. The SCSI bus requires termination at each end of the bus cable on all signal lines. These are 220R to +5 volts and 330R to 0 volts. Where no internal drive is fitted, termination is provided internally by a plug-on terminator PCB assembly, which is mechanically polarised. Power to these termination resistors is provided via diode D1, to allow target devices on the SCSI bus to power them should the initiator be switched off. The initiator may also power the terminators at the far end of the SCSI bus cable. Fuse FS1 limits the current to a maximum of 1 Amp. TR1 provides an open-collector drive to the SCSI reset signal when the SEC is used in initiator mode.

The SCSI expansion card state machine

When the ARM system memory clock is run at a different speed from that of the I/O clock, a period of synchronization (minimum 1 I/O clock cycle) is required at the beginning and end of each I/O cycle. These extra clock cycles cause the earlier SEC design to relinquish and re-arbitrate for SRAM access on every register transfer of an STM or LDM command, degrading potential performance. The solution to this was to cause the Issue 2+ state machine to hold access to the SRAM for the ARM for a number of clock cycles after the completion of the I/O cycle. This allows for synchronisation clock cycles and will, conveniently, span video DMA interruptions too. This is achieved by the use of a three bit counter built in to the RWPAL and count decode logic in the new PAL ADDPLUS. Figure 1 shows two accesses to the SBIC. The first access is a write to the address register in order to pre-select a register. The second is a register read. Note that because LA13 is high the second access is an E-cycle, even though the ARM has control. Figure 2 shows an LDM from SRAM. Note that LA13 is low throughout this command. When the extended cycle is complete the RW_DN signal is activated and the counter starts to count from zero again. However, each time an IORQ is received, it is reset to zero. Hence we see the counter oscillating between zero and one until the end of the LDM, when it counts out to seven, and the bus control is relinquished. Figure 3 shows an STM split up by video DMA accesses and the counter reaching a higher count before being reset to zero.

Figure 1

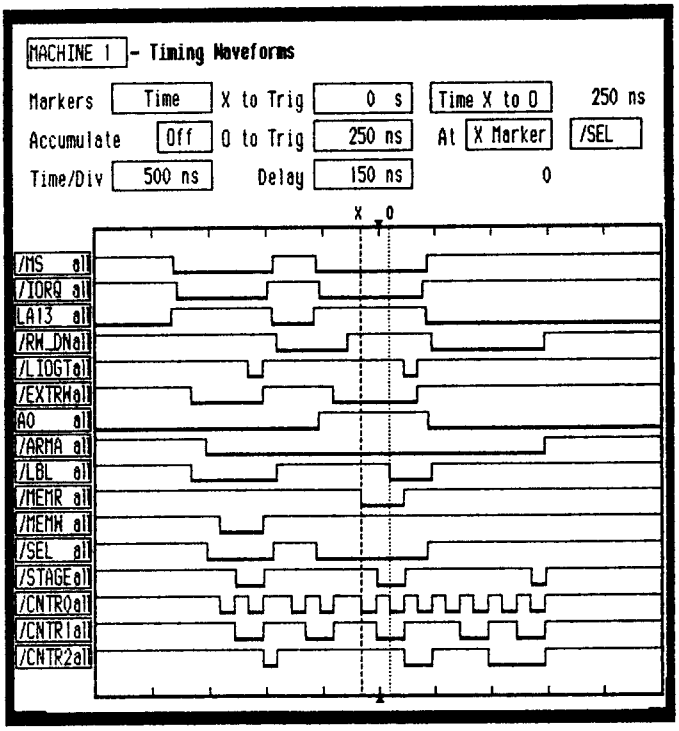


Figure 2

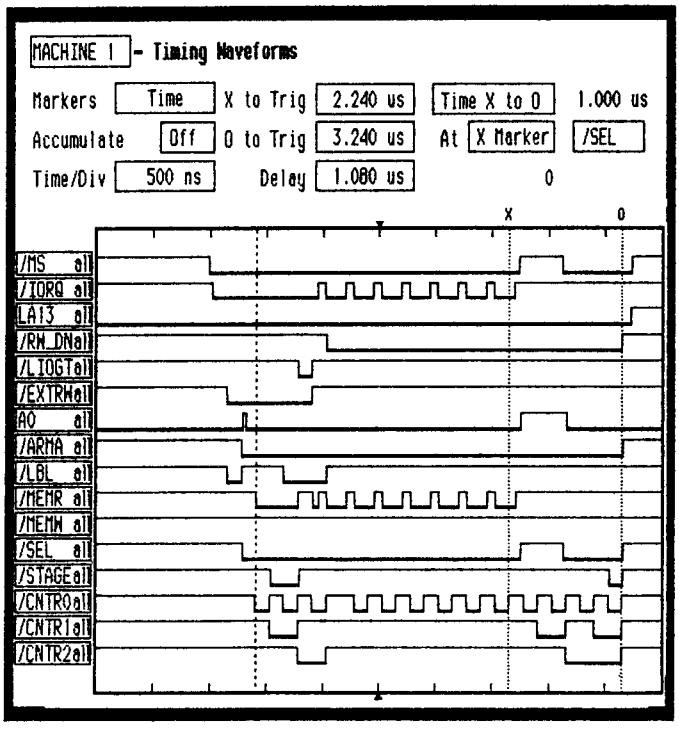
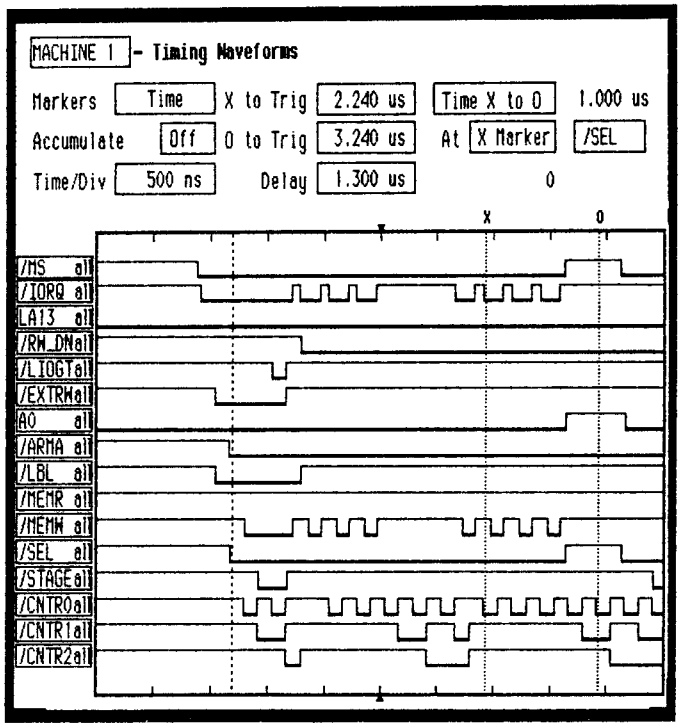


Figure 3


MACHINE 1 - Timing Waveforms
Markers Time X to Trig 2.240 us Time X to O 1.000 us
Accumulate Off O to Trig 3.240 us At X Marker /SEL
Time/Div 500 ns Delay 1.300 us O
X
O
/MS all
/IORQ all
LA13 all
/RW_DN all
/LIOGT all
/EXTRW all
A0 all
/ARMA all
/LBL all
/MEMR all
/MEMW all
/SEL all
/STAGE all
/CNTR0 all
/CNTR1 all
/CNTR2 all

Glossary of terms for PAL signal names

SCSI	Small computer systems interface
SBIC	SCSI bus interface controller
DMAC	Direct memory access controller
SRAM	Static random access memory
PCLK8M	8MHz clock
CLRINT	Clear interrupt
INTE	Interrupt enable
TC	Terminal count
INTRQ	Interrupt request
ARMA	ARM processor access
MS	Module select
URST	User reset
RST	Reset
ABE	ARM bus enable
IRST	Internal reset
IRQ	Interrupt request
DINT	DMA interrupt
FIQ	Fast interrupt
SINT	SCSI interrupt
SRST	SCSI reset
PIRQ	Podule interrupt
PRE	Podule read enable
PWE	Podule write enable
LA12	Latched address 12
LA13	Latched address 13
PS	Podule select
DACK	DMA acknowledge
A0	Address line 0
A23	Address line 23
EPRM	EPROM
SRLO	Static RAM low
SRHI	Static RAM high
PAGE	Page register
INTRD	Interrupt read
AEN	Address enable
IORQ	Input/output request
HLDRQ	Hold request
STAGE	Move to next stage
PNRW	Podule not read, write
BL	Buffer latch
LBL	Latched buffer latch
LIOGT	Latched IOGT
REL	Release
EXTRW	Extended read write
HLDAK	Hold acknowledge
IOGT	Input output grant
RA7	SRAM address 7
REF8M	Reference 8MHz clock
MEMW	Memory write
C2	Counter bit 2
C1	Counter bit 1
C0	Counter bit 0
IOWR	I/O device write
IORD	I/O device read
I/O	Input/Output
MEMR	Memory read
CNTR0	Counter bit 0
CNTR1	Counter bit 1
CNTR2	Counter bit 2
RWD	Read write done
B***	Buffered 'signal'
UDE	Upper data enable
NC	Not connected

Bibliography

- WD33C93A SCSI Bus Interface Controller Data Book (document no. 79-000199). Available from Western Digital (UK) Ltd, The Old Manor House, 17 West Street, Epsom, Surrey KT18 7RL.
- NEC Microprocessor and Peripheral Data Book, covering the uPD71071 DMA controller. Available from NEC Electronics (UK) Ltd, Cygnus House, Linford Wood Business Centre, Sunrise Parkway, Linford Wood, Milton Keynes MK14 6NP.
- 'A Series Podules' - a specification of the Acorn podule bus, available as an Application Note from Customer Services (address as at the front of this manual).
- Acorn *SCSI Expansion Card User Guide*, supplied with the SCSI expansion card.

Risc os	2.01 Eproms	2.5 Eproms
CH23	1 2 3	(1 2) 3
CH24	1 2 3	1 2 7
CH26	1 2 3	1 2 3
CH25	1	1 2
CH27		1 2

Risc OS 2.00 ROMS
23. 24. 26 all (23)
CH25 (1 2)
27 (1 2)

Part 3 - Links, plugs and sockets

Main PCB Links

Link	Fitted	Effect	Default
LK1	Yes	Internal auxiliary video connector providing access to the following signals: P1 Red P2 Green P3 Blue P4 H/CSync P5 VSync/Mode P6 0V	None
LK2	Yes	Provides option for video CSync to be superimposed on the Green video output signal: P1 Sync source P2 Green P3 0V Note that for a monitor that uses Sync on Green, move this link to the 1-2 position, and set the sync type to 1.	Shunt 2-3 (normal sync)
LK3	Yes	Used to configure the signal on pin 5 of the video socket: P1 VSync or $\overline{\text{VSync}}$ P2 Video socket pin 5 P3 Mode Note: Mode is a control signal used by some monitors with a SCART interface.	Shunt 2-3 (mode)
LK4	Yes	Test connector used in conjunction with Acorn designed test equipment: P1 5V P2 D<0> P3 La<21> P4 $\overline{\text{RomCS}}$ P5 $\overline{\text{Rst}}$ P6 0V	None
LK5	Yes	Provides simple oscillator signal.	None
LK6	Yes	Used to configure the signal on pin 4 of the video socket: P1 HSync/HSync P2 Video socket pin 4 P3 CSync	Shunt 2-3 (CSync)
LK7	No	Test point for non-volatile memory/RTC battery voltage: P1 0V P2 1.2V ± 0.2V	None
LK8	No	Test point for RTC clock frequency: P1 0V P2 32.768 KHz	None
LK9	Yes	Internal auxiliary audio connector providing access to the following signals: P1 Unfiltered Left P2 0V P3 Left Phones P4 0V P5 Aux Input P6 0V P7 Right Phones P8 0V P9 Unfiltered Right P10 0V	None
LK10	No	Internal test link, allowing phase adjustment of Hi-Res dot clock.	Trk 1-2
LK11	No	Internal test link, allowing phase adjustment of CK24M.	Trk 1-2
LK12	No	Allows the +5V supply to the floppy to be via the data cable or through a separate feed. For supply via data cable, cut track 22-3 and link positions 1-2.	Trk 2-3 (separate feed)
LK13	Yes	Connection point for the internal speaker: P1 0V P2 Signal	None
LK14	Yes	Connection point for POWER front panel LED.	None
LK15	Yes	Genlock connection point: P1 Internal clock, P2 Clock to VIDC, P3 Sink, P4 0V, P5 NC, P6 NC, P7 $\overline{\text{Ved0}}$, P8 $\overline{\text{Ved1}}$, P9 $\overline{\text{Ved2}}$, P10 $\overline{\text{Ved3}}$, P11 $\overline{\text{Supremacy}}$, P12 $\overline{\text{Hi}}$, P13 $\overline{\text{HSync}}$, P14 $\overline{\text{Vs}}$, P15 0V, P16 0V	Shunt 1-2 Shunt 3-4
LK23	Yes	Used in conjunction with LK24 and LK26 to select ROM SIZE: SIZE (Bits) LK24 LK23 LK26 512K 2-3 2-3 2-3 1M 2-3 2-3 2-3 2M 1-2 2-3 2-3 4M 1-2 1-2 2-3 8M 1-2 1-2 1-2	Shunts 2-3
LK24	Yes	See LK23.	
LK25	Yes	Used in conjunction with LK27 to select ROM TYPE: TYPE LK25 LK27 Non JEDEC 512K 1-2 1-2 Non JEDEC 1MB ROM 1-2 1-2 Non JEDEC 1MB EPROMs 1-2 1-2 JEDEC 1, 2, 4, and 8MB ROMs and EPROMs 1 -------1, 2 -----2	Shunts 1-2
LK26	Yes	See LK23.	None
LK27	Yes	See LK25.	None
LK28	Yes	Not allocated	
LK29	Yes	Test point for MEMC reference clocks P1 5V P2 RefW (MEMCw) P3 RefX (MEMCx) P4 RefY (MEMCy) P5 RefZ (MEMCz) P6 0V **NOTE: Links LK16 to LK22 inclusive are not fitted.**	None

Plugs

Plug	Fitted	Function/Specification
PL1	Yes	Serial Port. (IBM PC-AT Pinout) 9-way D-type plug. Pin Signal Pin Signal 1 DCD 6 DSR 2 RxD 7 RTS 3 TxD 8 CTS 4 DTR 9 RI 5 0V
PL2	Yes	Floppy disc power connector. P1 +5V P2 0V P3 0V P4 +12V
PL3	Yes	Processor module plug. This is a 96-way DIN 41612 plug. It provides all signals for CPU card.
PL4	Yes	Floppy Disc Drive Data Connector. This is a 34-way Box Header containing all the signals required by the internal floppy disc drive. Pin Signal Pin Signal 2 Dcirq* 20 Step* 4 Inuse* 22 Writedata* 6 Sel(3)* 24 Writegate* 8 Index* 26 Track00* 10 Sel(0)* 28 Writeprot* 12 Sel(1)* 30 Readdata* 14 Sel(2)* 32 Side1* 16 Motoron* 34 Ready* 18 Dirin* * All signals are active low. 1,3,13,15,17,19,21,23,25,27,29,31,33 all 0V 5,7,9,11 optional power lines. See LK12.
PL5	Yes	Power supply connector. This is a 6-way socket for connecting to the DC power supply. P1 +5V P2 0V P3 +12V P4 +5V P5 0V P6 -5V
PL6 to PL9		Optional power supply connectors. PL6 -5V PL7 +12V PL8 0V PL9 +5V
PL10 PL11		Chassis connection points. These two faston connectors provide a connection point between the keyboard cable screen and the case of the computer.

Sockets

Skt	Fitted	Function/Specification
SK1	Yes	Stereo headphone output. This is a 3-way 3.5mm stereo jack socket providing output to "Walkman-type" 32 ohm stereo headphones.
SK2	Yes	RGB video socket. This is a 9-way D-type socket providing an interface to RGB monitors and Scart TVs. Links 2, 3 and 6 can be used to alter the synchronisation signals to suit a variety of monitors RGB video levels are 0.7V Pk-Pk into 75 Ohm. Sync voltage levels are >= 2.0V (TTL). Pin Signal (IBM PC PGA pinning) 1 Red 2 Green 3 Blue 4 HSync 5 VSync/Mode 6,7,8,9 0V
SK3	Yes	Parallel printer port. 25-way D-type socket providing a parallel printer interface. Pin Signal Pin Signal Pin Signal 1 Stb 8 Pd(6) 15 nc 2 Pd(0) 9 Pd(7) 16 nc 3 Pd(1) 10 Ack 17-25 0V 4 Pd(2) 11 Bsy 5 Pd(3) 12 nc 6 Pd(4) 13 nc 7 Pd(5) 14 nc
SK4	Yes	Econet socket. 5-way DIN socket for connection to Econet LAN. Note, this is an upgrade. Pin Signal 1 Data 2 0V 3 Clock 4 Data 5 Clock
SK5, SK6, SK7	Yes	Memory expansion sockets. Allow memory expansion of 4MB at a time, up to a maximum of 16MB. Note, the DRAM cards must be inserted in the correct order: SK5, SK6, SK7.
SK8	Yes	Econet upgrade module socket. 5-way header used in conjunction with SK10. This module is identical to that used on Acorn Master series and Archimedes computers.
SK9	Yes	Backplane socket. Systems are normally supplied with a 4-way backplane already installed.
SK10	Yes	Econet upgrade module socket. 17-way header used in conjunction with SK8 to provide the electrical connection point for the internal Econet upgrade module. This module is identical to that used on Acorn Master series and Archimedes computers.

Sockets (cont.)

Skt	Fitted	Function/Specification
SK11	Yes	6-way mini-DIN socket providing the connection point for the keyboard. If required, a standard Archimedes keyboard may be plugged into this socket.
SK12	Yes	High resolution mono video output. Provides a 0.7V mono video signal (into 75 Ohm) at a dot rate of 96MHz. This requires a High resolution monitor to be connected.
SK13	Yes	High resolution mono vertical sync. Provides composite/vertical synchronisation pulses for the high resolution mono output.
SK14	Yes	High resolution mono horizontal sync. Provides horizontal synchronisation pulses for the high resolution mono output.

Internal expansion

Interface

Introduction

The computer supports an expansion card (podule) interface. The maximum power available per slot can be calculated from the following:

- The +5V supply rail is rated at a maximum of1A
- The +12V supply rail is rated at a maximum of 250mA
- The -5V supply rail is rated at a maximum of 50mA

Refer to the application note 'A Series Podules' for a full podule interface specification, available on request from Acorn Computers.

Pin	a	c	Description
1	0V	0V	Ground
2	LA[15]	–5V	
3	LA[14]	0V	Ground
4	LA[13]	0V	Ground
5	LA[12]	reserved	
6	LA[11]	MS[0]*	MEMC Podule select
7	LA[10]	reserved	
8	LA[9]	reserved	
9	LA[8]	reserved	
10	LA[7]	reserved	
11	LA[6]	reserved	
12	LA[5]	RST*	Reset (see note below)
13	LA[4]	PR/W*	Read/not write
14	LA[3]	PWE*	Write strobe
15	LA[2]	PRE*	Read strobe
16	BD[15]	PIRQ*	Normal interrupt
17	BD[14]	PFIQ*	Fast interrupt
18	BD[13]	S[6]*	
19	BD[12]	C1	I^2C serial bus clock
20	BD[11]	C0	I^2C serial bus data
21	BD[10]	S[7]*	External Podule select
22	BD[9]	PS[0]*	Simple Podule select
23	BD[8]	IOGT*	MEMC Podule handshake
24	BD[7]	IORQ*	MEMC Podule request
25	BD[6]	BL*	I/O data latch control
26	BD[5]	0V	Supply
27	BD[4]	CLK2	2MHz Synchronous clock
28	BD[3]	CLK8	8MHz Synchronous clock
29	BD[2]	REF8M	8MHz Reference clock
30	BD[1]	+5V	Supply
31	BD[0]	reserved	
32	+5V	+12V	

Note: The RST* signal is the system reset signal, driven by IOC on power up or by the keyboard reset switch. It is an open-collector signal, and expansion cards *may* drive it also if this is desirable. The pulse width should be at least 50ms.

Part 4 - Parts lists

The parts lists in this chapter detail the components used in the manufacture of workstations and upgrades.

The parts lists are given under the following headings:

- Main PCB
- 4MB RAM upgrade card
- Backplane
- ARM3 daughter card (PGA)
- *Keyboard and adaptor card (membrane keyboard)
- *Keyboard assembly (keyswitch keyboard)
- *Ethernet I card
- *Ethernet II card
- SCSI interface card.

*Note that either one of two different assemblies may be fitted.

There is a circuit diagram for each of the above items. All the circuit diagrams are included at the back of this manual.

Contact the Spares Department of Acorn Computers Limited (account holders only), or its authorised dealers and Approved Service Centres, for information as to which parts are available as spares.

Main PCB assembly parts list

Item	Description	Qty
1	BARE PCB	1
2	PCB ASSEMBLY DWG (1 per batch)	1
3	PCB CIRCUIT DIAGRAM (1 per batch)	1
7	MAIN PCB REAR PANEL	1
12	CONR 2W SHUNT 0.1" fitted to LK2, 3, 6, 15(x2), 23 - 27	10
13	WIRE 22SWG CPR TIN A/R (X1, X2, X3)	
14	WIRE 25SWG CPR TIN (X3)	
15	LABEL SERIAL PCB	1
16	FOAM PAD (11x24mm) (BT1)	1
21	SKT IC 20/0.3" SUPA (IC21)	1
22	SKT IC 20/0.3" SUPA (IC39)	1
23	SKT IC 32/0.6" SUPA (IC47)	1
24	SKT IC 32/0.6" SUPA (IC48)	1
25	SKT IC 32/0.6" SUPA (IC49)	1
26	SKT IC 32/0.6" SUPA (IC50)	1
27	SKT IC 68P PLCC (IC58)	1
28	SKT IC 68P PLCC (IC60)	1
29	SKT IC 68P PLCC (IC64)	1
30	SKT IC 20/0.3" SUPA (IC66)	1
31	SKT IC 20/0.3" SUPA (IC71)	1
BT1	BAT NI-CAD 1V2 280MAH PCB	1
C1	CPCTR 10U TANT 10V 20% 5P	1
C2	CPCTR 10U TANT 10V 20% 5P	1
C3	CPCTR 10U TANT 10V 20% 5P	1
C4	CPCTR 10U TANT 10V 20% 5P	1
C5	CPCTR 220U ALEC 16V RAD	1
C6	CPCTR 220U ALEC 16V RAD	1
C7	CPCTR 47U ALEC 16V RAD	1
C8	CPCTR 4U7 ALEC 16V RAD	1
C9	CPCTR 47U ALEC 16V RAD	1
C10	CPCTR 100U ALEC 25V RAD	1
C11	CPCTR 10U ALEC 16V RAD	1
C12	CPCTR 10U ALEC 16V RAD	1
C13	CPCTR 220U ALEC 16V RAD	1
C14	CPCTR 4U7 ALEC 16V RAD	1
C15	CPCTR 100U ALEC 25V RAD	1
C16	CPCTR 47U ALEC 16V RAD	1
C17	CPCTR 100U ALEC 25V RAD	1
C18	CPCTR 10U ALEC 16V RAD	1
C19	CPCTR 220U ALEC 16V RAD	1
C20	CPCTR 220U ALEC 16V RAD	1
C21	CPCTR 220U ALEC 16V RAD	1
C22	CPCTR 47N CER 30V 80%	1
C23	CPCTR 47U ALEC 16V RAD	1
C24	CPCTR 220U ALEC 16V RAD	1
C25	CPCTR 47N CER 30V 80%	1
C26	CPCTR 47U ALEC 16V RAD	1
C27	CPCTR 220U ALEC 16V RAD	1
C28	CPCTR 47N CER 30V 80%	1
C29	CPCTR 47U ALEC 16V RAD	1
C30	CPCTR 220U ALEC 16V RAD	1
C31	CPCTR 10U ALEC 16V RAD	1
C32	CPCTR 47N CER 30V 80%	1
C33	CPCTR 47U ALEC 16V RAD	1
C34	CPCTR 10U ALEC 16V RAD	1
C35	CPCTR 220U ALEC 16V RAD	1
C36	CPCTR 10U ALEC 16V RAD	1
C37	CPCTR 220U ALEC 16V RAD	1
C38	CPCTR 33/47N DCPLR 0.2"	1
C39	CPCTR 22N MPSTR 50V 10%	1
C40	CPCTR 22N MPSTR 50V 10%	1
C41	CPCTR 100N MPSTR 50V 10%	1
C42	CPCTR 100N MPSTR 50V 10%	1
C43	CPCTR 100N DCPLR SMD1210	1
C44	CPCTR 100P CPLT 30V 2%	1
C45	CPCTR 470P CPLT 30V 10%	1
C46	CPCTR 100N DCPLR SMD1210	1
C47	CPCTR 470P CPLT 30V 10%	1
C48	CPCTR 18P CPLT 30V 2%	1
C49	CPCTR 100P CPLT 30V 2%	1
C50	CPCTR 2N2 CPLT 30V 10% 5P	1
C51	CPCTR 2N2 CPLT 30V 10% 5P	1
C52	CPCTR 33N DCPLR SMD1210	1
C53	CPCTR 2N2 CPLT 30V 10% 5P	1
C54	CPCTR 2N2 CPLT 30V 10% 5P	1
C55	CPCTR 100P CPLT 30V 2%	1

Item	Description	Qty
C56	CPCTR 2N2 CPLT 30V 10% 5P	1
C57	CPCTR 33P CPLT 30V 2%	1
C58	CPCTR 470P CPLT 30V 10%	1
C59		NF
C60	CPCTR 100P CPLT 30V 2%	1
C61	CPCTR 100P CPLT 30V 2%	1
C62	CPCTR 100P CPLT 30V 2%	1
C63	CPCTR 470P CPLT 30V 10%	1
C64		NF
C65	CPCTR 1N CPLT 30V 10%	1
C66	CPCTR 100P CPLT 30V 2%	1
C67		NF
C68	CPCTR 100P CPLT 30V 2%	1
C69	CPCTR 27P CPLT 30V 2%	1
C70	CPCTR 2N2 CPLT 30V 10% 5P	1
C71	CPCTR 1N CPLT 30V 10%	1
C72	CPCTR 15P CPLT 30V 2%	1
C73		NF
C74	CPCTR 100P CPLT 30V 2%	1
C75	CPCTR 18P CPLT 30V 2%	1
C76		NF
C77	CPCTR 1N CPLT 30V 10%	1
C78	CPCTR 47P CPLT 30V 2%	1
C79	CPCTR 1N CPLT 30V 10%	1
C80	CPCTR 100P CPLT 30V 2%	1
C81		NF
C82	CPCTR 1N CPLT 30V 10%	1
C83	CPCTR 47P CPLT 30V 2%	1
C84	CPCTR 1N CPLT 30V 10%	1
C85	CPCTR 100P CPLT 30V 2%	1
C86	CPCTR 1N CPLT 30V 10%	1
C87	CPCTR 47P CPLT 30V 2%	1
C88	CPCTR 1N CPLT 30V 10%	1
C89	CPCTR 1N CPLT 30V 10%	1
C90	CPCTR 1N CPLT 30V 10%	1
C91	CPCTR 1N CPLT 30V 10%	1
C92	CPCTR 100P CPLT 30V 2%	1
C93	CPCTR 1N CPLT 30V 10%	1
C94		NF
C95	CPCTR 100P CPLT 30V 2%	1
C96	CPCTR 1N CPLT 30V 10%	1
C97		NF
C98	CPCTR 100P CPLT 30V 2%	1
C99		NF
C100	CPCTR 100P CPLT 30V 2%	1
C101	CPCTR 100P CPLT 30V 2%	1
C102	CPCTR 33N DCPLR SMD1210	1
C103	CPCTR 33N DCPLR SMD1210	1
C104	CPCTR 33N DCPLR SMD1210	1
C105	CPCTR 33N DCPLR SMD1210	1
C106	CPCTR 33N DCPLR SMD1210	1
C107	CPCTR 33N DCPLR SMD1210	1
C108	CPCTR 33N DCPLR SMD1210	1
C109	CPCTR 33N DCPLR SMD1210	1
C110	CPCTR 33N DCPLR SMD1210	1
C111	CPCTR 33N DCPLR SMD1210	1
C112	CPCTR 33N DCPLR SMD1210	1
C113	CPCTR 33N DCPLR SMD1210	1
C114	CPCTR 33N DCPLR SMD1210	1
C115	CPCTR 33N DCPLR SMD1210	1
C116	CPCTR 33N DCPLR SMD1210	1
C117	CPCTR 33N DCPLR SMD1210	1
C118	CPCTR 33N DCPLR SMD1210	1
C119	CPCTR 33N DCPLR SMD1210	1
C120	CPCTR 33N DCPLR SMD1210	1
C121	CPCTR 33N DCPLR SMD1210	1
C122	CPCTR 33N DCPLR SMD1210	1
C123	CPCTR 33N DCPLR SMD1210	1
C124	CPCTR 33N DCPLR SMD1210	1
C125	CPCTR 33N DCPLR SMD1210	1
C126	CPCTR 33N DCPLR SMD1210	1
C127	CPCTR 33N DCPLR SMD1210	1
C128	CPCTR 33N DCPLR SMD1210	1
C129	CPCTR 33N DCPLR SMD1210	1
C130	CPCTR 33N DCPLR SMD1210	1
C131	CPCTR 33N DCPLR SMD1210	1
C132	CPCTR 33N DCPLR SMD1210	1
C133	CPCTR 33N DCPLR SMD1210	1
C134	CPCTR 33N DCPLR SMD1210	1
C135	CPCTR 33N DCPLR SMD1210	1
C136	CPCTR 33N DCPLR SMD1210	1
C137	CPCTR 33N DCPLR SMD1210	1

Item	Description	Qty
C138	CPCTR 33N DCPLR SMD1210	1
C139	CPCTR 33N DCPLR SMD1210	1
C140	CPCTR 33N DCPLR SMD1210	1
C141	CPCTR 33N DCPLR SMD1210	1
C142	CPCTR 33N DCPLR SMD1210	1
C143	CPCTR 33N DCPLR SMD1210	1
C144	CPCTR 33N DCPLR SMD1210	1
C145	CPCTR 33N DCPLR SMD1210	1
C146	CPCTR 33N DCPLR SMD1210	1
C147	CPCTR 33N DCPLR SMD1210	1
C148	CPCTR 33N DCPLR SMD1210	1
C149	CPCTR 33N DCPLR SMD1210	1
C150	CPCTR 100N DCPLR SMD1210	1
C151	CPCTR 100N DCPLR SMD1210	1
C152	CPCTR 100N DCPLR SMD1210	1
C153	CPCTR 100N DCPLR SMD1210	1
C154	CPCTR 100N DCPLR SMD1210	1
C155	CPCTR 100N DCPLR SMD1210	1
C156	CPCTR 100N DCPLR SMD1210	1
C157	CPCTR 100N DCPLR SMD1210	1
C158	CPCTR 100N DCPLR SMD1210	1
C159	CPCTR 33N DCPLR SMD1210	1
C160	CPCTR 100N DCPLR SMD1210	1
C161	CPCTR 100N DCPLR SMD1210	1
C162	CPCTR 100N DCPLR SMD1210	1
C163	CPCTR 33N DCPLR SMD1210	1
C164	CPCTR 100N DCPLR SMD1210	1
C165	CPCTR 100N DCPLR SMD1210	1
C166	CPCTR 100N DCPLR SMD1210	1
C167	CPCTR 33N DCPLR SMD1210	1
C168	CPCTR 100N DCPLR SMD1210	1
C169	CPCTR 100N DCPLR SMD1210	1
C170	CPCTR 100N DCPLR SMD1210	1
C171	CPCTR 33N DCPLR SMD1210	1
C172	CPCTR 100N DCPLR SMD1210	1
C173	CPCTR 100N DCPLR SMD1210	1
C174	CPCTR 100N DCPLR SMD1210	1
C175	CPCTR 100N DCPLR SMD1210	1
C176	CPCTR 100N DCPLR SMD1210	1
C177	CPCTR 100N DCPLR SMD1210	1
C178	CPCTR 100N DCPLR SMD1210	1
C179	CPCTR 100N DCPLR SMD1210	1
C180	CPCTR 100N DCPLR SMD1210	1
C181	CPCTR 100N DCPLR SMD1210	1
C182	CPCTR 100N DCPLR SMD1210	1
C183	CPCTR 100N DCPLR SMD1210	1
C184	CPCTR 33N DCPLR SMD1210	1
C185	CPCTR 33N DCPLR SMD1210	1
C186	CPCTR 10U TANT 10V 20% 5P	1
C187	CPCTR 10U TANT 10V 20% 5P	1
C188	CPCTR 10U TANT 10V 20% 5P	1
C189	CPCTR 10U TANT 10V 20% 5P	1
C190	CPCTR 10U TANT 10V 20% 5P	1
C191	CPCTR 10U TANT 10V 20% 5P	1
C192	CPCTR 10U TANT 10V 20% 5P	1
C193	CPCTR 10U TANT 10V 20% 5P	1
C194	CPCTR 10U TANT 10V 20% 5P	1
C195	CPCTR 10U TANT 10V 20% 5P	1
C196		NF
C197		NF
C198	CPCTR 33N DCPLR SMD1210	1
D1	DIODE SI 1N4005 600V 1A	1
D2	DIODE SI1N4148	1
D3	DIODE SI 1N4148	1
D4	DIODE SI 1N4148	1
D5	DIODE SI 1N4148	1
D6	DIODE SI 1N4148	1
D7	DIODE SI 1N4148	1
D8	DIODE SI 1N4148	1
D9	DIODE BAT85 SBL	1
D10	DIODE SI 1N418	1
D11	DIODE SI 1N418	1
D12	DIODE SI 1N418	1
D13	DIODE SI 1N418	1
D14	DIODE SI 1N418	1
D15	DIODE SI 1N418	1
FS1	FUSE 2AO F AX LEAD LBC	1
IC1	IC 74HCT14 CMOS 14/0.3"	1
IC2	IC 65C51 ACIA CMOS 2MHZ	1
IC3	IC 74HCT14 CMOS 14/0.3"	1
IC4	IC 74ACT174 CMOS 16/0.3"	1
IC5	IC 75189 RS232 RCVR	1
IC6	IC 75189 RS232 RCVR	1

Item	Description	Qty
IC7	IC 74ACT245 CMOS 20/0.3"	1
IC8	IC 74ACT153 CMOS 16/0.3"	1
IC9	IC 74HCT573 CMOS 20/0.3"	1
IC10	IC 74HCT573 CMOS 20/0.3"	1
IC11	IC DRAM 1MX1 20ZIP 80NS	1
IC12	IC DRAM 1MX1 20ZIP 80NS	1
IC13	IC DRAM 1MX1 20ZIP 80NS	1
IC14	IC DRAM 1MX1 20ZIP 80NS	1
IC15	IC DRAM 1MX1 20ZIP 80NS	1
IC16	IC DRAM 1MX1 20ZIP 80NS	1
IC17	IC 26LS30 RS422/423 DRVR	1
IC18	IC DRAM 1MX1 20ZIP 80NS	1
IC19	IC DRAM 1MX1 20ZIP 80NS	1
IC20	IC DRAM 1MX1 20ZIP 80NS	1
IC21	IO SLOW PAL {0760,203}	1
IC22	IC 8583 RTC RAM 8/0.3"	1
IC23	IC DRAM 1MX1 20ZIP 80NS	1
IC24	IC DRAM 1MX1 20ZIP 80NS	1
IC25	IC DRAM 1MX1 20ZIP 80NS	1
IC26	IC DRAM 1MX1 20ZIP 80NS	1
IC27	IC DRAM 1MX1 20ZIP 80NS	1
IC28	IC DRAM 1MX1 20ZIP 80NS	1
IC29	IC DRAM 1MX1 20ZIP 80NS	1
IC30	IC DRAM 1MX1 20ZIP 80NS	1
IC31	IC DRAM 1MX1 20ZIP 80NS	1
IC32	IC DRAM 1MX1 20ZIP 80NS	1
IC33	IC DRAM 1MX1 20ZIP 80NS	1
IC34	IC DRAM 1MX1 20ZIP 80NS	1
IC35	IC DRAM 1MX1 20ZIP 80NS	1
IC36	IC DRAM 1MX1 20ZIP 80NS	1
IC37	IC DRAM 1MX1 20ZIP 80NS	1
IC38	IC DRAM 1MX1 20ZIP 80NS	1
IC39	MMEMC ADD PAL {0760,203}	1
IC40	IC DRAM 1MX1 20ZIP 80NS	1
IC41	IC DRAM 1MX1 20ZIP 80NS	1
IC42	IC DRAM 1MX1 20ZIP 80NS	1
IC43	IC DRAM 1MX1 20ZIP 80NS	1
IC44	IC DRAM 1MX1 20ZIP 80NS	1
IC45	IC DRAM 1MX1 20ZIP 80NS	1
IC46	IC DRAM 1MX1 20ZIP 80NS	1
IC47	RISC OS 2.01 ROM1	1
IC48	RISC OS 2.01 ROM2	1
IC49	RISC OS 2.01 ROM3	1
IC50	RISC OS 2.01 ROM4	1
IC51	IC 74AC04 CMOS 14/0.3	1
IC52	IC 74AC04 CMOS 14/0.3	1
IC53	IC 74AC11 CMOS 14/0.3	1
IC54	IC 74S00 TTL 14/0.3	1
IC55	IC 7406 TTL 14/0.3"	1
IC56	IC 74AS21 TTL 14/0.3"	1
IC57	IC 74AS74 TTL 14/0.3	1
IC58	IC IOC PLSTC	1
IC59	IC 74AC32 CMOS 14/0.3"	1
IC60	IC MEMC1A 12MHZ PLSTC	1
IC61	IC 74F166 FAST 16/0.3	1
IC62	IC 74HC00 CMOS 14/0.3"	1
IC63	IC 74AC86 CMOS 14/0.3	1
IC64	IC VIDC 1A PLSTC	1
IC65	IC 74HC175 CMOS 16/0.3"	1
IC66	MEMC FAST PAL {0760,203}	1
IC67	IC 74HC573 CMOS 20/0.3"	1
IC68	IC 74HC573 CMOS 20/0.3"	1
IC69	IC 74HC573 CMOS 20/0.3"	1
IC70	IC 1772 FDC 28/0.6	1
IC71	MEMC SYNC PAL {0760,203}	1
IC72	IC 74ACT74 CMOS 14/0.3	1
IC73	IC 74AC574 CMOS 20/0.3	1
IC74	IC 74HCT573 CMOS 20/0.3"	1
IC75	IC 74HC138 CMOS 16/0.3"	1
IC76	IC 74HCT573 CMOS 20/0.3"	1
IC77	IC 74HC574 CMOS 20/0.3"	1
IC78	IC LM324 QUAD OP AMP	1
IC79	IC 74LS374 TTL 20/0.3"	1
IC80	IC LM386 AUDIO AMP	1
L1	CHOKE RF 2U2H AX Q=30	1
L2	CHOKE 80OHM/100MHZ	1
L3	CHOKE 80OHM/100MHZ	1
L4	CHOKE 80OHM/100MHZ	1
L5	CHOKE 80OHM/100MHZ	1
L6	CHOKE 80OHM/100MHZ	1
L7	CHOKE 80OHM/100MHZ	1
L8	CHOKE 80OHM/100MHZ	1
L9	CHOKE 80OHM/100MHZ	1

Item	Description	Qty
L10	CHOKE 80OHM/100MHZ	1
L11	CHOKE 80OHM/100MHZ	1
L12	CHOKE 80OHM/100MHZ	1
L13	CHOKE 80OHM/100MHZ	1
L14	CHOKE 80OHM/100MHZ	1
L15	CHOKE 80OHM/100MHZ	1
L16	CHOKE 80OHM/100MHZ	1
L17	CHOKE RF U22H AX Q=35 7X3	1
L18	CHOKE RF 33UH AX Q=45	1
LK1	CONR 6W WAFR 0.1" ST PCB	1
LK2	CONR 3W WAFR 0.1" ST PCB	1
LK3	CONR 3W WAFR 0.1" ST PCB	1
LK4	CONR 6W WAFR 0.1" ST PCB	1
LK5	CONR 2W WAFR 0.1" ST PCB	1
LK6	CONR 3W WAFR 0.1" ST PCB	1
LK7		NF
LK8		NF
LK9	CONR 10W WAFR 2ROW 0.1"	1
LK10		NF
LK11		NF
LK12		NF
LK13	CONR 2W WAFR 0.1" ST LK	1
LK14	CONR 2W WAFR 0.1" ST LK	1
LK15	CONR 16W WAFR 2ROW 0.1"	1
LK23	CONR 3W WAFR 0.1" ST PCB	1
LK24	CONR 3W WAFR 0.1" ST PCB	1
LK25	CONR 2W WAFR 0.1" ST PCB	1
LK26	CONR 3W WAFR 0.1" ST PCB	1
LK27	CONR 2W WAFR 0.1" ST PCB	1
LK29	CONR 6W WAFR 0.1" ST PCB	1
PL1	CONRD 9WPLG RA PCB+RFI+L	1
PL2	CONR 4W PLG PCB ST DISC P	1
PL3	CONR 96W PLG ST PCB REV	1
PL4	CONR 34W BOX IDC LP ST	1
PL5	CONR 6W PLG PCB DCPWR	1
PL6		NF
PL7		NF
PL8		NF
PL9		NF
PL10	FSTN TAB 6,3MMX0,8 ST PCB	1
PL11	FSTN TAB 6,3MMX0,8 ST PCB	1
Q1	TRANS BF689K NPN TO92 .1"	1
Q2	TRANS 2N3904 NPN TO92 .2"	1
Q3	TRANS 2N3906 PNP TO92 .2"	1
Q4	TRANS BC239C NPN TO92 .2"	1
Q5	TRANS 2N3906 PNP TO92 .2"	1
Q6	TRANS 2N3906 PNP TO92 .2"	1
Q7	TRANS 2N3906 PNP TO92 .2"	1
Q8	TRANS 2N3906 PNP TO92 .2"	1
Q10	TRANS 2N3906 PNP TO92 .2"	1
Q12	TRANS BC239C NPN TO92 .2"	1
Q13	TRANS BC239C NPN TO92 .2"	1
Q14	TRANS BC239C NPN TO92 .2"	1
R1	RES 4K7 SMD 5% 0W25 1206	1
R2	RES 4K7 SMD 5% 0W25 1206	1
R3	RES 4K7 SMD 5% 0W25 1206	1
R4	RES 4K7 SMD 5% 0W25 1206	1
R5	RES 4K7 SMD 5% 0W25 1206	1
R6	RES 4K7 SMD 5% 0W25 1206	1
R7	RES 4K7 SMD 5% 0W25 1206	1
R8	RES 33K SMD 5% 0W25 1206	1
R9	RES 1K2 SMD 5% 0W25 1206	1
R10	RES 10K SMD 5% 0W25 1206	1
R11	RES 10R SMD 5% 0W25 1206	1
R12	RES 22R SMD 5% 0W25 1206	1
R13	RES 1K2 SMD 5% 0W25 1206	1
R14	RES 10K SMD 5% 0W25 1206	1
R15	RES 220R SMD 5% 0W25 1206	1
R16	RES 4K7 SMD 5% 0W25 1206	1
R17	RES 1K0 SMD 5% 0W25 1206	1
R18	RES 2K2 SMD 5% 0W25 1206	1
R19	RES 4K7 SMD 5% 0W25 1206	1
R20	RES 330R SMD 5% 0W25 1206	1
R21	RES 6K8 SMD 5% 0W25 1206	1
R22	RES 68R SMD 5% 0W25 1206	1
R23	RES 1K0 SMD 5% 0W25 1206	1
R24	RES 180R SMD 5% 0W25 1206	1
R25	RES 100K SMD 5% 0W25 1206	1
R26	RES 1K0 SMD 5% 0W25 1206	1
R27	RES 1K0 SMD 5% 0W25 1206	1
R28	RES 220R SMD 5% 0W25 1206	1
R29	RES 1K0 SMD 5% 0W25 1206	1
R30	RES 68R SMD 5% 0W25 1206	1

Item	Description	Qty
R31	RES 33R SMD 5% 0W25 1206	1
R32	RES 1K0 SMD 5% 0W25 1206	1
R33	RES 180R SMD 5% 0W25 1206	1
R34	RES 100K SMD 5% 0W25 1206	1
R35	RES 100R SMD 5% 0W25 1206	1
R36	RES 1K0 SMD 5% 0W25 1206	1
R37	RES 10K SMD 5% 0W25 1206	1
R38	RES 10K SMD 5% 0W25 1206	1
R39	RES 56R SMD 5% 0W25 1206	1
R40	RES 100K SMD 5% 0W25 1206	1
R41	RES 68R SMD 5% 0W25 1206	1
R42	RES 100K SMD 5% 0W25 1206	1
R43	RES 4K7 SMD 5% 0W25 1206	1
R44	RES 1K0 SMD 5% 0W25 1206	1
R45	RES 47K SMD 5% 0W25 1206	1
R46	RES 100K SMD 5% 0W25 1206	1
R47	RES 33R SMD 5% 0W25 1206	1
R48	RES 68R SMD 5% 0W25 1206	1
R49	RES 68R SMD 5% 0W25 1206	1
R50	RES 33R SMD 5% 0W25 1206	1
R51	RES 1K0 SMD 5% 0W25 1206	1
R52	RES 10K SMD 5% 0W25 1206	1
R53	RES 10K SMD 5% 0W25 1206	1
R54	RES 33R SMD 5% 0W25 1206	1
R55	RES 3K3 SMD 5% 0W25 1206	1
R56	RES 68R SMD 5% 0W25 1206	1
R57	RES 33R SMD 5% 0W25 1206	1
R58	RES 68R SMD 5% 0W25 1206	1
R59	RES 1K0 SMD 5% 0W25 1206	1
R60	RES 100K SMD 5% 0W25 1206	1
R61	RES 100K SMD 5% 0W25 1206	1
R62	RES 2K2 SMD 5% 0W25 1206	1
R63	RES 68R SMD 5% 0W25 1206	1
R64	RES 68R SMD 5% 0W25 1206	1
R65	RES 68R SMD 5% 0W25 1206	1
R66	RES 68R SMD 5% 0W25 1206	1
R67	RES 33R SMD 5% 0W25 1206	1
R68	RES 68R SMD 5% 0W25 1206	1
R69	RES 1K0 SMD 5% 0W25 1206	1
R70	RES 10K SMD 5% 0W25 1206	1
R71	RES 2K2 SMD 5% 0W25 1206	1
R72	RES 56R SMD 5% 0W25 1206	1
R73	RES 3K3 SMD 5% 0W25 1206	1
R74	RES 68R SMD 5% 0W25 1206	1
R75	RES 10K SMD 5% 0W25 1206	1
R76	RES 680R SMD 5% 0W25 1206	1
R77	RES 1K0 SMD 5% 0W25 1206	1
R78	RES 10K SMD 5% 0W25 1206	1
R79	RES 100K SMD 5% 0W25 1206	1
R80	RES 330R SMD 5% 0W25 1206	1
R81	RES 68R SMD 5% 0W25 1206	1
R82	RES 1K0 SMD 5% 0W25 1206	1
R83	RES 22R SMD 5% 0W25 1206	1
R84	RES 68R SMD 5% 0W25 1206	1
R85	RES 68R SMD 5% 0W25 1206	1
R86	RES 68R SMD 5% 0W25 1206	1
R87	RES 33R SMD 5% 0W25 1206	1
R88	RES 33R SMD 5% 0W25 1206	1
R89	RES 33R SMD 5% 0W25 1206	1
R90	RES 1K0 SMD 5% 0W25 1206	1
R91	RES 1K0 SMD 5% 0W25 1206	1
R92	RES 56R SMD 5% 0W25 1206	1
R93	RES 33R SMD 5% 0W25 1206	1
R94	RES 3K3 SMD 5% 0W25 1206	1
R95	RES 68R SMD 5% 0W25 1206	1
R96	RES 68R SMD 5% 0W25 1206	1
R97	RES 330R SMD 5% 0W25 1206	1
R98	RES 1K0 SMD 5% 0W25 1206	1
R99	RES 100K SMD 5% 0W25 1206	1
R100	RES 100K SMD 5% 0W25 1206	1
R101	RES 220R SMD 5% 0W25 1206	1
R102	RES 68R SMD 5% 0W25 1206	1
R103	RES 1K0 SMD 5% 0W25 1206	1
R104	RES 22R SMD 5% 0W25 1206	1
R105	RES 68R SMD 5% 0W25 1206	1
R106	RES 68R SMD 5% 0W25 1206	1
R107	RES 68R SMD 5% 0W25 1206	1
R108	RES 33R SMD 5% 0W25 1206	1
R109	RES 33R SMD 5% 0W25 1206	1
R110	RES 33R SMD 5% 0W25 1206	1
R111	RES 330R SMD 5% 0W25 1206	1
R112	RES 33R SMD 5% 0W25 1206	1
R113	RES 33K SMD 5% 0W25 1206	1

Item	Description	Qty
R114	RES 3K3 SMD 5% 0W25 1206	1
R115	RES 3R3 SMD 5% 0W25 1206	1
R116	RES 68R SMD 5% 0W25 1206	1
R117	RES 4K7 SMD 5% 0W25 1206	1
R118	RES 100K SMD 5% 0W25 1206	1
R119	RES 100K SMD 5% 0W25 1206	1
R120	RES 68R SMD 5% 0W25 1206	1
R121	RES 68R SMD 5% 0W25 1206	1
R122	RES 22R SMD 5% 0W25 1206	1
R123	RES 4K7 SMD 5% 0W25 1206	1
R124	RES 68R SMD 5% 0W25 1206	1
R125	RES 68R SMD 5% 0W25 1206	1
R126	RES 68R SMD 5% 0W25 1206	1
R127	RES 33R SMD 5% 0W25 1206	1
R128	RES 33R SMD 5% 0W25 1206	1
R129	RES 68R SMD 5% 0W25 1206	1
R130	RES 1K0 SMD 5% 0W25 1206	1
R131	RES 10K SMD 5% 0W25 1206	1
R132	RES 3K3 SMD 5% 0W25 1206	1
R133	RES 3R3 SMD 5% 0W25 1206	1
R134	RES 68R SMD 5% 0W25 1206	1
R135	RES 4K7 SMD 5% 0W25 1206	1
R136	RES 100K SMD 5% 0W25 1206	1
R137	RES 100K SMD 5% 0W25 1206	1
R138	RES 68R SMD 5% 0W25 1206	1
R139	RES 68R SMD 5% 0W25 1206	1
R140	RES 22R SMD 5% 0W25 1206	1
R141	RES 4K7 SMD 5% 0W25 1206	1
R142	RES 68R SMD 5% 0W25 1206	1
R143	RES 68R SMD 5% 0W25 1206	1
R144	RES 68R SMD 5% 0W25 1206	1
R145	RES 68R SMD 5% 0W25 1206	1
R146	RES 33R SMD 5% 0W25 1206	1
R147	RES 33R SMD 5% 0W25 1206	1
R148	RES 68R SMD 5% 0W25 1206	1
R149	RES 10K SMD 5% 0W25 1206	1
R150	RES 3K3 SMD 5% 0W25 1206	1
R151	RES 330R SMD 5% 0W25 1206	1
R152	RES 68R SMD 5% 0W25 1206	1
R153	RES 100K SMD 5% 0W25 1206	1
R154	RES 100K SMD 5% 0W25 1206	1
R155	RES 68R SMD 5% 0W25 1206	1
R156	RES 68R SMD 5% 0W25 1206	1
R157	RES 22R SMD 5% 0W25 1206	1
R158	RES 47K SMD 5% 0W25 1206	1
R159	RES 68R SMD 5% 0W25 1206	1
R160	RES 68R SMD 5% 0W25 1206	1
R161	RES 68R SMD 5% 0W25 1206	1
R162	RES 68R SMD 5% 0W25 1206	1
R163	RES 33R SMD 5% 0W25 1206	1
R164	RES 33R SMD 5% 0W25 1206	1
R165	RES 4K7 SMD 5% 0W25 1206	1
R166	RES 10K SMD 5% 0W25 1206	1
R167	RES 4K7 SMD 5% 0W25 1206	1
R168	RES 68R SMD 5% 0W25 1206	1
R169	RES 100K SMD 5% 0W25 1206	1
R170	RES 68R SMD 5% 0W25 1206	1
R171	RES 68R SMD 5% 0W25 1206	1
R172	RES 22R SMD 5% 0W25 1206	1
R173	RES 4K7 SMD 5% 0W25 1206	1
R174	RES 68R SMD 5% 0W25 1206	1
R175	RES 68R SMD 5% 0W25 1206	1
R176	RES 68R SMD 5% 0W25 1206	1
R177	RES 68R SMD 5% 0W25 1206	1
R178	RES 33R SMD 5% 0W25 1206	1
R179	RES 33R SMD 5% 0W25 1206	1
R180	RES 33R SMD 5% 0W25 1206	1
R181	RES 33R SMD 5% 0W25 1206	1
R182	RES 10K SMD 5% 0W25 1206	1
R183	RES 4K7 SMD 5% 0W25 1206	1
R184	RES 68R SMD 5% 0W25 1206	1
R185	RES 68R SMD 5% 0W25 1206	1
R186	RES 22R SMD 5% 0W25 1206	1
R187	RES 4K7 SMD 5% 0W25 1206	1
R188	RES 68R SMD 5% 0W25 1206	1
R189	RES 68R SMD 5% 0W25 1206	1
R190	RES 68R SMD 5% 0W25 1206	1
R191	RES 68R SMD 5% 0W25 1206	1
R192	RES 33R SMD 5% 0W25 1206	1
R193	RES 68R SMD 5% 0W25 1206	1
R194	RES 33R SMD 5% 0W25 1206	1
R195	RES 68R SMD 5% 0W25 1206	1
R196	RES 22R SMD 5% 0W25 1206	1

Item	Description	Qty
R197	RES 4K7 SMD 5% 0W25 1206	1
R198	RES 68R SMD 5% 0W25 1206	1
R199	RES 68R SMD 5% 0W25 1206	1
R200	RES 68R SMD 5% 0W25 1206	1
R201	RES 68R SMD 5% 0W25 1206	1
R202	RES 33R SMD 5% 0W25 1206	1
R203	RES 22R SMD 5% 0W25 1206	1
R204	RES 4K7 SMD 5% 0W25 1206	1
R205	RES 68R SMD 5% 0W25 1206	1
R206	RES 33R SMD 5% 0W25 1206	1
R207	RES 1K2 SMD 5% 0W25 1206	1
R208	RES 10K SMD 5% 0W25 1206	1
R209	RES 1K0 SMD 5% 0W25 1206	1
R210	RES 1K0 SMD 5% 0W25 1206	1
R211	RES 1K0 SMD 5% 0W25 1206	1
R212	RES 43R2 MF 1% 0W25 E96	1
R213	RES 43R2 MF 1% 0W25 E96	1
R214	RES 22K1 MF 1% 0W25 E96	1
R215	RES 22K1 MF 1% 0W25 E96	1
R216	RES 22K1 MF 1% 0W25 E96	1
R217	RES 220R SMD 5% 0W25 1206	1
R218	RES 43R2 MF 1% 0W25 E96	1
R219	RES 22K1 MF 1% 0W25 E96	1
R220	RES 220R SMD 5% 0W25 1206	1
R221	RES 100K SMD 5% 0W25 1206	1
R222	RES 22K1 MF 1% 0W25 E96	1
R223	RES 22K1 MF 1% 0W25 E96	1
R224	RES 150R SMD 5% 0W25 1206	1
R225	RES 100K SMD 5% 0W25 1206	1
R226	RES 1K00 MF 1% 0W25 E96	1
R227	RES 22K1 MF 1% 0W25 E96	1
R228	RES 330R SMD 5% 0W25 1206	1
R229	RES 100K SMD 5% 0W25 1206	1
R230	RES 332R MF 1% 0W25 E96	1
R231	RES 1K00 MF 1% 0W25 E96	1
R232	RES 330R SMD 5% 0W25 1206	1
R233	RES 100K SMD 5% 0W25 1206	1
R234	RES 332R MF 1% 0W25 E96	1
R235	RES 1K00 MF 1% 0W25 E96	1
R236	RES 150R SMD 5% 0W25 1206	1
R237	RES 33K SMD 5% 0W25 1206	1
R238	RES 332R MF 1% 0W25 E96	1
R239	RES 1K00 MF 1% 0W25 E96	1
R240	RES 150R SMD 5% 0W25 1206	1
R241	RES 33K SMD 5% 0W25 1206	1
R242	RES 22K1 MF 1% 0W25 E96	1
R243	RES 1K8 SMD 5% 0W25 1206	1
R244	RES 560R SMD 5% 0W25 1206	1
R245	RES 10K SMD 5% 0W25 1206	1
R246	RES 100K SMD 5% 0W25 1206	1
R247	RES 100K SMD 5% 0W25 1206	1
R248	RES 100K SMD 5% 0W25 1206	1
R249	RES 100K SMD 5% 0W25 1206	1
R250	RES 10R SMD 5% 0W25 1206	1
R251	RES 1K0 SMD 5% 0W25 1206	1
R252	RES 1K0 SMD 5% 0W25 1206	1
R253	RES 1K0 SMD 5% 0W25 1206	1
R254	RES 1K0 SMD 5% 0W25 1206	1
R255	RES 22R SMD 5% 0W25 1206	1
R256	RES 33R SMD 5% 0W25 1206	1
R257	RES 22R SMD 5% 0W25 1206	1
R258	RES 22R SMD 5% 0W25 1206	1
R259	RES 22R SMD 5% 0W25 1206	1
R260	RES 22R SMD 5% 0W25 1206	1
R261	RES 33R SMD 5% 0W25 1206	1
R262	RES 33R SMD 5% 0W25 1206	1
R263	RES 33R SMD 5% 0W25 1206	1
R264	RES 33R SMD 5% 0W25 1206	1
R265	RES 33R SMD 5% 0W25 1206	1
R266	RES 33R SMD 5% 0W25 1206	1
R267	RES 33R SMD 5% 0W25 1206	1
R268	RES 33R SMD 5% 0W25 1206	1
R269	RES 33R SMD 5% 0W25 1206	1
R270	RES 33R SMD 5% 0W25 1206	1
R271	RES 33R SMD 5% 0W25 1206	1
R272	RES 33R SMD 5% 0W25 1206	1
R273	RES 33R SMD 5% 0W25 1206	1
R274	RES 33R SMD 5% 0W25 1206	1
R275	RES 68R SMD 5% 0W25 1206	1
R276	RES 33R SMD 5% 0W25 1206	1
R277	RES 33R SMD 5% 0W25 1206	1
R278	RES 33R SMD 5% 0W25 1206	1
R279	RES 22R SMD 5% 0W25 1206	1

Item	Description	Qty
R280	RES 33R SMD 5% 0W25 1206	1
R281	RES 33R SMD 5% 0W25 1206	1
R282	RES 33R SMD 5% 0W25 1206 00	1
R283	RES 33R SMD 5% 0W25 1206	1
R284	RES 33R SMD 5% 0W25 1206	1
R285	RES 33R SMD 5% 0W25 1206	1
R286	RES 33R SMD 5% 0W25 1206	1
R287	RES 68R SMD 5% 0W25 1206	1
R288	RES 1K0 SMD 5% 0W25 1206	1
R289	RES 1K0 SMD 5% 0W25 1206	1
R290	RES 33R SMD 5% 0W25 1206	1
SK1	CONR JKSKT 3W 3,5MM RAPCB	1
SK2	CONRD 9WSKT RA PCB+RFI+L	1
SK3	CONRD 25W SKT RAPCB+RFI+L	1
SK4	CONR 5W SKT DIN SCRN PCB	1
SK5	CONR 96W SKT ST ABC PCB	1
SK6	CONR 96W SKT ST ABC PCB	1
SK7	CONR 96W SKT ST ABC PCB	1
SK8	CONR 5W SKT HSNG 0.1" PCB	1
SK9	CONR 96W SKT ST ABC PCB	1
SK10	CONR 17W SKT HSNG .1" PCB	1
SK11	SKT 6W MINDIN RA PCB RFI	1
SK12	CONR BNC SKT RAPCB METAL	1
SK13	CONR BNC SKT RAPCB METAL	1
SK14	CONR BNC SKT RAPCB METAL	1
X1	XTAL 96MHZ HC18 5TH O/T	1
X2	XTAL 1.8432MHZ HC18	1
X3	XTAL 32.768KHZ CC 0.05P	1
X4	XTAL OSC 72.00MHZ 14/0.3"	1
X5		NF
X6	XTAL OSC 36.00MHZ 14/0.3"	1
X7	XTAL OSC 25.175MHZ 14/0.3	1

4MB RAM card (optional upgrade)

Item	Description	Qty
1	BARE PCB	1
2	PCB ASSEMBLY DRG	1
3	PCB CIRCUIT DIAGRAM	1
15	LABEL SERIAL PCB	1
C1	CPCTR 47U TANT SMD	1
C2	CPCTR 47U TANT SMD	1
C3	CPCTR 47U TANT SMD	1
C4	CPCTR 47U TANT SMD	1
C5	CPCTR 33N DCPLR SMD1210	1
C6	CPCTR 33N DCPLR SMD1210	1
C7	CPCTR 33N DCPLR SMD1210	1
C8	CPCTR 33N DCPLR SMD1210	1
C9	CPCTR 33N DCPLR SMD1210	1
C10	CPCTR 33N DCPLR SMD1210	1
C11	CPCTR 33N DCPLR SMD1210	1
C12	CPCTR 33N DCPLR SMD1210	1
C13	CPCTR 33N DCPLR SMD1210	1
C14	CPCTR 33N DCPLR SMD1210	1
IC1	IC 1MX4 DRAM 80NS SOJ	1
IC2	IC 1MX4 DRAM 80NS SOJ	1
IC3	IC 1MX4 DRAM 80NS SOJ	1
IC4	IC 1MX4 DRAM 80NS SOJ	1
IC5	IC 1MX4 DRAM 80NS SOJ	1
IC6	IC 1MX4 DRAM 80NS SOJ	1
IC7	IC 1MX4 DRAM 80NS SOJ	1
IC8	IC 1MX4 DRAM 80NS SOJ	1
IC9	IC 74AC04 CMOS 14P SMD	1
IC10	IC MEMC1A 12MHZ PLSTC	1
L1	CHOKE 80R/100MHZ SMD	1
L2	CHOKE 80R/100MHZ SMD	1
L3	CHOKE 80R/100MHZ SMD	1
PL1	CONR 96W PLG RA ABC PCB	1
R1	RES 68R SMD 5% 0W25 1206	1
R2	RES 68R SMD 5% 0W25 1206	1
R3	RES 68R SMD 5% 0W25 1206	1
R4	RES 68R SMD 5% 0W25 1206	1
R5	RES 68R SMD 5% 0W25 1206	1
R6	RES 68R SMD 5% 0W25 1206	1
R7	RES 68R SMD 5% 0W25 1206	1
R8	RES 68R SMD 5% 0W25 1206	1
R9	RES 68R SMD 5% 0W25 1206	1
R10	RES 68R SMD 5% 0W25 1206	1
R11	RES 68R SMD 5% 0W25 1206	1
R12	RES 68R SMD 5% 0W25 1206	1
R13	RES 68R SMD 5% 0W25 1206	1
R14	RES 68R SMD 5% 0W25 1206	1
R15	RES 68R SMD 5% 0W25 1206	1
R16	RES 68R SMD 5% 0W25 1206	1
R17	RES 68R SMD 5% 0W25 1206	1
R18	RES 68R SMD 5% 0W25 1206	1
R19	RES 68R SMD 5% 0W25 1206	1
R20	RES 68R SMD 5% 0W25 1206	1
R21	RES 68R SMD 5% 0W25 1206	1
R22	RES 68R SMD 5% 0W25 1206	1
R23	RES 68R SMD 5% 0W25 1206	1
R24	RES 68R SMD 5% 0W25 1206	1
R25	RES 68R SMD 5% 0W25 1206	1
R26	RES 68R SMD 5% 0W25 1206	1
R27	RES 68R SMD 5% 0W25 1206	1
R28	RES 68R SMD 5% 0W25 1206	1
R29	RES 68R SMD 5% 0W25 1206	1
R30	RES 68R SMD 5% 0W25 1206	1
R31	RES 68R SMD 5% 0W25 1206	1
R32	RES 68R SMD 5% 0W25 1206	1
R33	RES 33R SMD 5% 0W25 1206	1
R34	RES 33R SMD 5% 0W25 1206	1
R35	RES 33R SMD 5% 0W25 1206	1
R36	RES 33R SMD 5% 0W25 1206	1
R37	RES 33R SMD 5% 0W25 1206	1
R38	RES 33R SMD 5% 0W25 1206	1
R39	RES 33R SMD 5% 0W25 1206	1
R40	RES 33R SMD 5% 0W25 1206	1

Notes on MEMC:

MEMCs **MUST** be Acorn Part Number 2201,393, to ensure correct timing parameters.

To allow for future expansion, PL1 pins A25,16 and 8 should be left open circuit, not connected to +5V. This change will be carried out on any future issue of the PCB.

Backplane adaptor

Item	Description	Qty
1	BARE PCB	1
2	PCB ASSEMBLY DRG (1 PER BATCH)	1
3	PCB CIRCUIT DIAGRAM (1 PER BATCH)	1
15	LABEL SERIAL PCB 15x50mm	1
C1	CPCTR 33/47N DCPLR 0.2"	1
C2	CPCTR 33/47N DCPLR 0.2"	1
C3	CPCTR 33/47N DCPLR 0.2"	1
C4	CPCTR 47U ALEC 16V AX	1
C5	CPCTR 47U ALEC 16V AX	1
C6	CPCTR 47U ALEC 16V AX	1
IC1	IC 74HC139 CMOS 16/0.3"	1
IC2	BP INT MASK PAL(0760003)	1
IC3	BP INT MASK PAL(0760003)	1
PL1	CONR 96W PLG RA ABC PCB	1
R1	RES 10K C/MF 5% 0W25	1
R2	RES 10K C/MF 5% 0W25	1
R3	RES 10K C/MF 5% 0W25	1
R4	RES 10K C/MF 5% 0W25	1
SK1	CONR 64W SKT ST AC PCB SH	1
SK2	CONR 64W SKT ST AC PCB SH	1
SK3	CONR 64W SKT ST AC PCB SH	1
SK4	CONR 64W SKT ST AC PCB SH	1

ARM3 (PGA) Daughter card

Item	Description	Qty
1	BARE PCB	1
2	PCB ASSEMBLY DRG	1
3	PCB CIRCUIT DIAGRAM	1
15	LABEL SERIAL PCB	1
C1	CPCTR 47U ALEC 10V AX	1
C2	CPCTR 47U ALEC 10V AX	1
C3	CPCTR 22P CPLT 30V 2%	1
C4	CPCTR 33N DCPLR SMD1210	1
C5	CPCTR 33N DCPLR SMD1210	1
C6	CPCTR 33N DCPLR SMD1210	1
C7	CPCTR 33N DCPLR SMD1210	1
IC1	ARM3 CPU {PGA}	1
IC2	IC 74ACT74 CMOS 14/0.3	1
R1	RES 10K SMD 5% 0W25 1206	1
R2	RES 68R SMD 5% 0W25 1206	1
R4	RES 100R SMD 5% 0W25 1206	1
R5	RES 22R SMD 5% 0W25 1206	1
R6	RES 22R SMD 5% 0W25 1206	1
R7	RES 22R SMD 5% 0W25 1206	1
R8	RES 22R SMD 5% 0W25 1206	1
R9	RES 22R SMD 5% 0W25 1206	1
R10	RES 22R SMD 5% 0W25 1206	1
R11	RES 22R SMD 5% 0W25 1206	1
R12	RES 22R SMD 5% 0W25 1206	1
R13	RES 22R SMD 5% 0W25 1206	1
R14	RES 22R SMD 5% 0W25 1206	1
R15	RES 22R SMD 5% 0W25 1206	1
R16	RES 22R SMD 5% 0W25 1206	1
R17	RES 22R SMD 5% 0W25 1206	1
R18	RES 22R SMD 5% 0W25 1206	1
R19	RES 22R SMD 5% 0W25 1206	1
R20	RES 22R SMD 5% 0W25 1206	1
R21	RES 22R SMD 5% 0W25 1206	1
R22	RES 22R SMD 5% 0W25 1206	1
R23	RES 22R SMD 5% 0W25 1206	1
R24	RES 22R SMD 5% 0W25 1206	1
R25	RES 22R SMD 5% 0W25 1206	1
R26	RES 22R SMD 5% 0W25 1206	1
R27	RES 22R SMD 5% 0W25 1206	1
R28	RES 22R SMD 5% 0W25 1206	1
R29	RES 22R SMD 5% 0W25 1206	1
R30	RES 22R SMD 5% 0W25 1206	1
R31	RES 22R SMD 5% 0W25 1206	1
R32	RES 22R SMD 5% 0W25 1206	1
R33	RES 22R SMD 5% 0W25 1206	1
R34	RES 22R SMD 5% 0W25 1206	1
R35	RES 22R SMD 5% 0W25 1206	1
R36	RES 22R SMD 5% 0W25 1206	1
R37	RES 22R SMD 5% 0W25 1206	1
R38	RES 22R SMD 5% 0W25 1206	1
R39	RES 22R SMD 5% 0W25 1206	1
R40	RES 22R SMD 5% 0W25 1206	1
R41	RES 22R SMD 5% 0W25 1206	1
R42	RES 22R SMD 5% 0W25 1206	1
R43	RES 22R SMD 5% 0W25 1206	1
R44	RES 22R SMD 5% 0W25 1206	1
R45	RES 22R SMD 5% 0W25 1206	1
R46	RES 22R SMD 5% 0W25 1206	1
R47	RES 22R SMD 5% 0W25 1206	1
R48	RES 22R SMD 5% 0W25 1206	1
R49	RES 22R SMD 5% 0W25 1206	1
R50	RES 22R SMD 5% 0W25 1206	1
R51	RES 22R SMD 5% 0W25 1206	1
R52	RES 22R SMD 5% 0W25 1206	1
R53	RES 22R SMD 5% 0W25 1206	1
R54	RES 22R SMD 5% 0W25 1206	1
R55	RES 22R SMD 5% 0W25 1206	1
R56	RES 22R SMD 5% 0W25 1206	1
R57	RES 22R SMD 5% 0W25 1206	1
R58	RES 22R SMD 5% 0W25 1206	1
R59	RES 22R SMD 5% 0W25 1206	1
R60	RES 22R SMD 5% 0W25 1206	1
R61	RES 22R SMD 5% 0W25 1206	1
R62	RES 22R SMD 5% 0W25 1206	1
R63	RES 22R SMD 5% 0W25 1206	1
R64	RES 22R SMD 5% 0W25 1206	1
R65	RES 22R SMD 5% 0W25 1206	1
R66	RES 22R SMD 5% 0W25 1206	1
R67	RES 22R SMD 5% 0W25 1206	1
R68	RES 22R SMD 5% 0W25 1206	1
SK1	CONR 96W SKT RA PCB REV	1
X1	XTAL OSC 14/0.3"	1

Keyboard adaptor PCB (membrane keyboard)

Item	Description	Qty
1	BARE PCB	1
2	PCB ASSEMBLY DWG (1 PER BATCH)	1
3	PCB CIRCUIT DIAGRAM (1 PER BATCH)	1
9	KEYBOARD CABLE ASSEMBLY	1
16	LABEL SERIAL PCB	1
19	WIRE 25SWG CPR TIN (A/R X1)	
21	SKT IC 40/0.6" SUPA (IC3)	1
C1	CPCTR 1N CPLT 30V 10%	1
C2	CPCTR 1N CPLT 30V 10%	1
C3	CPCTR 1N CPLT 30V 10%	1
C4	CPCTR 1N CPLT 30V 10%	1
C5	CPCTR 47U ALEC 10V AX	1
C6	CPCTR 33/47N DCPLR 0.2"	1
C7	CPCTR 33/47N DCPLR 0.2"	1
C8	CPCTR 27P CPLT 30V 2%	1
C9	CPCTR 33/47N DCPLR 0.2"	1
C10	CPCTR 27P CPLT 30V 2%	1
C11	CPCTR 33/47N DCPLR 0.2"	1
C12	CPCTR 33/47N DCPLR 0.2"	1
C13	CPCTR 33/47N DCPLR 0.2"	1
C14	CPCTR 4U7 ALEC 10V AX	1
C15	CPCTR 47U ALEC 10V AX	1
C16	CPCTR 1N CPLT 30V 10%	1
C17	CPCTR 1N CPLT 30V 10%	1
C18	CPCTR 1N CPLT 30V 10%	1
C19	CPCTR 1N CPLT 30V 10%	1
C20	CPCTR 1N CPLT 30V 10%	1
C21	CPCTR 1N CPLT 30V 10%	1
C22	CPCTR 1N CPLT 30V 10%	1
C23	CPCTR 1N CPLT 30V 10%	1
IC1	IC 74HCT4051 CMOS 16/0.3"	1
IC2	IC 74HCT4051 CMOS 16/0.3"	1
IC3	IC KBD CTRLR {0708,051}	1
IC4	IC 74HC14 CMOS 14/0.3"	1
IC5	IC 74HCT4051 CMOS 16/0.3"	1
IC6	IC 74HCT14 CMOS 14/0.3"	1
L1	CHOKE 80OHM/100MHZ	1
LK1		NF
LK2		NF
LK3		NF
LK4		NF
LK5		NF
LK6		NF
LK7		NF
R1	RES 220R C/MF 5% 0W25	1
R2	RES 330R C/MF 5% 0W25	1
R3	RES 47K C/MF 5% 0W25	1
R4	RES 47K C/MF 5% 0W25	1
R5	RES 47K C/MF 5% 0W25	1
R6	RES 47K C/MF 5% 0W25	1
R7	RES 47K C/MF 5% 0W25	1
R8	RES 47K C/MF 5% 0W25	1
R9	RES 47K C/MF 5% 0W25	1
R10	RES 47K C/MF 5% 0W25	1
R11	RES 47K C/MF 5% 0W25	1
R12	RES 47K C/MF 5% 0W25	1
R13	RES 47K C/MF 5% 0W25	1
R14	RES 47K C/MF 5% 0W25	1
R15	RES 47K C/MF 5% 0W25	1
R16	RES 270R C/MF 5% 0W25	1
R17	RES 47K C/MF 5% 0W25	1
R18	RES 47K C/MF 5% 0W25	1
R19	RES 47K C/MF 5% 0W25	1
R20	RES 4K7 C/MF 5% 0W25	1
R21	RES 10K C/MF 5% 0W25	1
R22	RES 330R C/MF 5% 0W25	1
R23	RES 330R C/MF 5% 0W25	1
R24	RES 10K C/MF 5% 0W25	1
R25	RES 330R C/MF 5% 0W25	1
R26	RES 10K C/MF 5% 0W25	1
R27	RES 220R C/MF 5% 0W25	1
R28	RES 270R C/MF 5% 0W25	1
R29	RES 10K C/MF 5% 0W25	1
R30	RES 10K C/MF 5% 0W25	1
R31	RES 10K C/MF 5% 0W25	1
R32	RES 10K C/MF 5% 0W25	1
R33	RES 22K C/MF 5% 0W25	1
R34	RES 100R C/MF 5% 0W25	1
R35	RES 100R C/MF 5% 0W25	1
SK1	CONR 20W FLEX PCB	1

Keyboard adaptor PCB (cont) (membrane keyboard)

Item	Description	Qty
SK2	CONR 20W FLEX PCB	1
SK3	CONR 9W SKT M/DIN RA RFI	1
SW1	SW 2P MOM CO P/B RA PCB	1
X1	XTAL 12.00MHZ HC18	1

Keyboard assembly (keyswitch keyboard)

This is a service replacement item. Part numbers:

0186,012 (keyboard subassembly)

0086,900/A (complete keyboard unit).

Ethernet I

Item	Description	Qty
	BARE PCB	1
	ASSEMBLY DRAWING	1*
	CIRCUIT DIAGRAM	1*
IC13	IC GAL 1 {0760,200 TBP}	1
IC14	IC GAL 2 {0760,200 TBP}	1
IC15	IC GAL 3 {0760,200 TBP}	1
IC12	IC PROM {0702,719 TBP}	1
	ETHER/CHEAPERNET REAR PNL	1
	PCB SUPPORT MOUNTING BRKT	1
R18	RES 4K7 C/MF 5% 0W25	1
R12-15	RES 39R2 MF 1% 0W25 E96	4
R7-10	RES 43R2 MF 1% 0W25 E96	4
R6	RES 78R7 MF 1% 0W25 E96	1
R2,3,16	RES 243R MF 0%5 0W25	3
R11	RES 1M0 HIVOLT 5% 0W25	1
RP1	RES 10Kx5 NET SIL 6P 5%	1
C1,2	CPCTR CPLT 33p 30V 2%	2
C4,5,9, 10,15	CPCTR ALEC 47uF 16V RAD	5
C8	CPCTR CML 220n 25V 80%	1
C18,19	CPCTR CER 10n 100V 20%	2
C13	CPCTR MPSTR 22n 50V 10%	1
C3	CPCTR CLASY 10n 250V 20%	1
"A",C6, 7,14, 16,17	CPCTR DCPLR 33/47n 0.2"	12
IC10,11	IC 62256 SRAM 100nS 32Kx8	2
IC16	IC 82501 SIA NMOS 20/0/3"	1
IC17	IC 82502 TRAN MOS 16/0.3"	1
IC1	IC 82586 LAN NMOS 48/0.6"	1
IC2,3	IC 74HCT244 CMOS 20/0.3"	2
IC6,7, 8,9	IC 74HCT245 CMOS 20/0.3"	4
IC4,5	IC 74HCT573 CMOS 20/0.3"	2
IC18	IC 74ACT240 CMOS 20/0.3"	1
DC1	(was DC)DC/DC CONV 12V TO 5V,10V	1
Q1	(was TR1)TRANS BC239 NPN TO92 EBC	1
D1,2	DIODE IN4150 SI 50V DO35	2
LK10	CONR WAFR 3W 0.1" ST PCB	1
LK3-8, 10	CONR 2W SHUNT 0.1"	7
SKT	IC 16/03" SUPA USE ON IC17	1
SKT	IC 20/0.3" SUPA USE ON IC16	1
SKT	IC 48/0.6" SUPA USE ON IC1	1
SK1	(was PL2)CONR 15W SKT RA PCB+RFI	1
PL1	CONR 64W PLG RA AC PCB	1
LK3-8	CONR WAFR 6W 0.1" ST PCB	3
PL3	CONR BNC SKT PNL 50R INSU	1
	15W D SLIDE LOCK ASSY USE ON SK1	1
X1	XTAL 20MHZ HC18 20PF P/L	1
T1	ISO TRANS 16PIN DIL 0.3"	1
WIRE	22SWG CPR TIN USE ON X1,PL3	A/R
SCW	M2.5x6 PAN HD POSI USE ON 1,16,17	4
SCW	M3x10 PAN HD POSI USE ON 1,17,89	2
NUT	M2.5 STL FULL Z/PAS USE ON 108	2
NUT	M3 STL FULL Z/PAS USE ON 96,109	2
WSHR	M2.5 SPRF IT STL USE ON 108	2
WSHR	M3 SPRF IT STL USE ON 96,109	2
	*1 PER BATCH	

Ethernet II

Item	Description	Qty
1	BARE PCB	1
2	ASSEMBLY DRAWING	1*
3	CIRCUIT DIAGRAM	1*
7	ETHERNET II REAR PANEL	1
8	PCB SUPPORT MOUNTING BRKT	1
11	CONR 2W SHUNT 0.1"	6
	LK3 to 8	
13	SKT IC 20/0.3" SUPA	4
	IC10,14,18,19	
14	SKT IC 32/0.6" SUPA	1
	IC6	
15	SKT IC 68W PLCC	1
	IC3	
16	SKT IC 28W PLCC	1
	IC1	
18	15W D SLIDE LOCK ASSY	1
	SK1	
19		
20	WIRE 22SWG CPR TIN	A/R
	X1,SK2	
22	SCW M2,5x6 PAN HD POSI	4
	USE WITH ITEMS 1,7 & 8	
23	SCW M3x10 PAN HD POSI	2
	USE WITH ITEMS 1,8 & SK1	
25	NUT M2,5 STL FULL Z/PAS	2
	USED WITH ITEM 22	
26	NUT M3 STL FULL Z/PAS	2
	USED WITH ITEM 23	
28	WSHR M2,5 SPRF IT STL	4
	USED WITH ITEM 22	
29	WSHR M3 SPRF IT STL	2
	USED WITH ITEM 23	
R1	RES 39R2 MF 1% 0W25 E96	1
R2	RES 39R2 MF 1% 0W25 E96	1
R3	RES 68R C/MF 5% 0W25	1
R4	RES 39R2 MF 1% 0W25 E96	1
R5	RES 100K C/MF 5% 0W25	1
R6	RES 39R2 MF 1% 0W25 E96	1
R7	RES 1K00 MF 1% 0W25 E96	1
R8	RES 1K5 C/MF 5% 0W25	1
R9	RES 274R MF 1% 0W25 E96	1
R10	RES 220R C/MF 5% 0W25	1
R11	RES 1K5 C/MF 5% 0W25	1
R12	RES 274R MF 1% 0W25 E96	1
R13	RES 1K5 C/MF 5% 0W25	1
R14	RES 1M0 HIVOLT 5% 0W25	1
R15	RES 56R C/MF 5% 0W25	1
RP1	RESNET 1K5x7 SIL 8P 2%	1
RP2	RESNET 1K5x7 SIL 8P 2%	1
C1	CPCTR 100U ALEC 25V RAD	1
C2	CPCTR ALEC 47uF 16V RAD	1
C3	CPCTR ALEC 47uF 16V RAD	1
C4	CPCTR DCPLR 33/47n 0.2"	1
C5	CPCTR DCPLR 33/47n 0.2"	1
C6	CPCTR DCPLR 33/47n 0.2"	1
C7	CPCTR DCPLR 33/47n 0.2"	1
C8	CPCTR DCPLR 33/47n 0.2"	1
C9	CPCTR DCPLR 33/47n 0.2"	1
C10	CPCTR DCPLR 33/47n 0.2"	1
C11	CPCTR DCPLR 33/47n 0.2"	1
C12	CPCTR DCPLR 33/47n 0.2"	1
C13	CPCTR DCPLR 33/47n 0.2"	1
C14	CPCTR DCPLR 33/47n 0.2"	1
C15	CPCTR DCPLR 33/47n 0.2"	1
C16	CPCTR DCPLR 33/47n 0.2"	1
C17	CPCTR DCPLR 33/47n 0.2"	1
C18	CPCTR DCPLR 33/47n 0.2"	1
C19	CPCTR DCPLR 33/47n 0.2"	1
C20	CPCTR DCPLR 33/47n 0.2"	1
C21	CPCTR DCPLR 33/47n 0.2"	1
C22	CPCTR CLASY 10n 250V 20%	1
C23	CPCTR 10N CPLT 30V 80%	1
C24	CPCTR 10N CPLT 30V 80%	1
C25	CPCTR CPLT 33p 30V 2%	1
C26	CPCTR CPLT 10p 30V 2%	1
C27	CPCTR 10N CPLT 30V 80%	1
C28	CPCTR 10N CPLT 30V 80%	1
C29	CPCTR 10U TANT 16V 20%	1
IC1	IC 8391A MCC 28 PLCC	1

Item	Description	Qty
IC2	IC 8392A TRNSCVR 16/0.3"	1
IC3	IC 8390C NIC 68 PLCC	1
IC4	IC 74HC245 CMOS 20/0.3"	1
IC5	IC 74HC245 CMOS 20/0.3"	1
IC6	IC ROM {0727,128 TBP}	1
IC7	IC 74HCT273 CMOS 20/0.3"	1
IC8	IC 74HCT273 CMOS 20/0.3"	1
IC9	IC 74HCT573 CMOS 20/0.3"	1
IC10	IC PAL 1 {0760,200 TBP}	1
IC11	IC 74ACT646 CMOS 20/0.3"	1
IC12	IC 74HCT573 CMOS 20/0.3"	1
IC13	IC 74ACT646 CMOS 20/0.3"	1
IC14	IC PAL 2 {0760,200 TBP}	1
IC15	IC SRAM 32Kx8 100nS 28/.6	1
IC16	IC SRAM 32Kx8 100nS 28/.6	1
IC17	IC DC/DC CONV 5V TO -9V	1
IC18	IC PAL 3 {0760,200 TBP}	1
IC19	IC PAL 4 {0760,200 TBP}	1
IC20	IC 74HC04 CMOS 14/0.3"	1
IC21	IC 74HC04 CMOS 14/0.3"	1
D1	DIODE IN4150 SI 50V DO35	1
FS1	FUSE 500MA FF AX LEAD LBC	1
TF1	TXF ISO LAN 16/0.3"	1
LK1	NOT FITTED	
LK2	NOT FITTED	
LK3}		
LK4}		
LK5}	CONR 6W WAFR 0.1" ST PCB	3
LK6}		
LK7}		
LK8}		
LK9	NOT FITTED	
LK10	NOT FITTED	
LK11	NOT FITTED	
LK12	NOT FITTED	
LK13	NOT FITTED	
LK14	NOT FITTED	
LK15	NOT FITTED	
LK16	NOT FITTED	
SK1	CONR 15W SKT RA PCB +RFI	1
SK2	CONR BNC SKT RAPCB INSU	1
PL1	CONR 64W PLG RA AC PCB	1
X1	XTAL 20MHZ HC18 20PF P/L	1
	*1 PER BATCH	

SCSI interface card (issue 2)

Item	Description	Qty
1	BARE SCSI PCB {Iss 2+}	1
2	ASSEMBLY DRAWING	1*
3	CIRCUIT DIAGRAM	1*
6	SCSI PCB BACKPANEL	1
8	PODULE PCB BRACKET {STD}	2
10	CONRDL 50W PLG SCSI TERM SK1	1
13	CONR 2W SHUNT 0.1" LK4,5,6,7	4
15	SKT IC 20/0.3" SUPA IC9,12,14,15,18	5
16	SKT IC 32/0.6" SUPA IC5	1
17	SKT IC 44W PLCC IC16	1
18	SKT IC 52W PLCC IC17	1
21	SCW M2,5x6 PAN HD POSI USE ON ITEM 6	2
22	SCW M3x8 PAN HD POSI Z&P USE ON ITEM 1 AND SK1	2
24	NUT M3 STL FULL Z/PAS USE ON ITEM 22	2
25	WSHR M3 PLN STL Z/PAS USE ON ITEM 22	2
26	WSHR M2,5 PLN STL Z/PASS USE ON ITEM 21	2
28	RIVET POP DOME 3,2D & THK USE ON ITEMS 1 AND 8	2
R1	RES 270R C/MF 5% 0W25	1
R2	RES 68R C/MF 5% 0W25	1
R3	RES 10K C/MF 5% 0W25	1
R4	RES 10R C/MF 5% 0W25	1
R5	RES 10K C/MF 5% 0W25	1
R10	RES 4K7 C/MF 5% 0W25	1
R17	RES 1K5 C/MF 5% 0W25	1
R19	RES 470R C/MF 5% 0W25	1
R21	RES 10K C/MF 5% 0W25	1
R22	RES 10K C/MF 5% 0W25	1
R25	RES 10K C/MF 5% 0W25	1

Item	Description	Qty
C1	CPCTR 33/47n DCPLR 0.2"	1
C2	CPCTR 100u ALEC 25V 20%	1
TO	CPCTR 33/47n DCPLR 0.2"	17
TO	CPCTR 100P CPLT 30V 2%	12
TO	CPCTR 100P CPLT 30V 2%	5
IC1	IC 74HCT273 CMOS 20/0.3"	1
IC2	IC 74HC245 CMOS 20/0.3"	1
IC3	IC 74HCT541 CMOS 20/0.3"	1
IC4	IC 74HC245 CMOS 20/0.3"	1
IC5	IC SCSI EPROM {0727,256}	1
IC6	IC 74HC245 CMOS 20/0.3"	1
IC7	IC 74HCT573 CMOS 20/0.3"	1
IC8	IC 74HC245 CMOS 20/0.3"	1
IC9	IC GAL 2+ {0760,200 TBP}	1
IC10	IC 74HCT541 CMOS 20/0.3"	1
IC11	IC SRAM 32Kx8 100nS 28/0.6	1
IC12	IC GAL 4+ {0760,200 TBP}	1
IC13	IC SRAM 32Kx8 100nS 28/0.6	1
IC14	IC GAL 3+ {0760,200 TBP}	1
IC15	IC GAL 1+ {0760,200 TBP}	1
IC16	IC 33C93A SCSI 44PLCC	1
IC17	IC 71071 DMAC 52PLCC	1
IC18	IC GAL 5+ {0760,200 TBP}	1
TR1	TRANS BC239C NPN TO92 .2"	1
D1	DIODE BYV10-40 SCHOTTKY	1
LK1	CONR 2W WAFR 0.1" ST PCB	1
LK2	CONR 2W WAFR 0.1" ST PCB	1
LK3	CONR 2W WAFR 0.1" ST PCB	1
LK4	CONR 2W WAFR 0.1" ST PCB	1
LK5	CONR 2W WAFR 0.1" ST PCB	1
LK6	CONR 2W WAFR 0.1" ST PCB	1
LK7	CONR 2W WAFR 0.1" ST PCB	1
LK8	NOT FITTED	
LK9	NOT FITTED	
LK10	CONR 2W WAFR 0.1" ST PCB	1
SK1	CONRDL 50W SKT RA PCB M3	1
PL1	CONR 64W PLG RA AC PCB	1
PL2	CONR 50W BOX IDC LP ST	1
FS1	FUSE 1A0 F 8.7x2.54mm LBC	1
	*1 PER BATCH	

Chapter 201 Ethernet podule

20.2 Introduction

This chapter describes the hardware design and architecture of the Ethernet podule. This product provides users of the Acorn Technical Publishing System with a high performance network capability designed to conform to the IEEE 802.3 10base5 (Ethernet) and 10base2 (Cheapernet), 10 Mbps baseband standards.

Ethernet was developed by the Xerox Corporation in the early 1970s and a specification made available in 1980. This specification known as the 'Blue Book' was used as the basis for the IEEE and ECMA standards. All new equipment (including this product) is or should be designed to the IEEE standard. This allows interworking with existing Ethernet equipment, at least at the physical level.

The remainder of this document makes the following assumptions about the reader:

- A working knowledge of the Acorn podule bus is assumed. A guide to the Acorn podule bus is in Chapter 10, *Podules and backplane.*
- An understanding of the basic architecture of the Ethernet/IEEE 802.3 standard is assumed. The Intel publication *The LAN Components User's Manual* is particularly useful and contains a suitable introduction to local area network standards. It is recommended that the reader obtain a copy as reference to it is made in this document.

20.3 Specification

Acorn document 0373,000/PS is the product specification.

20.4 Basic operation and block diagram

The figure below is a block diagram of the Ethernet/Cheapernet podule.

The main functional blocks are:

- the net controller: Intel 82586 (LANCE)
- the serial interface adaptor: Intel 82501 (SIA)
- transceiver: Intel 82502
- attachment unit interface (AUI) socket (D-type)
- isolation transformers and power supply
- bus buffers and transceivers
- the RAM buffer
- the RAM page register
- a PROM based 'extended' podule ID
- the control register
- the PAL based state machine.

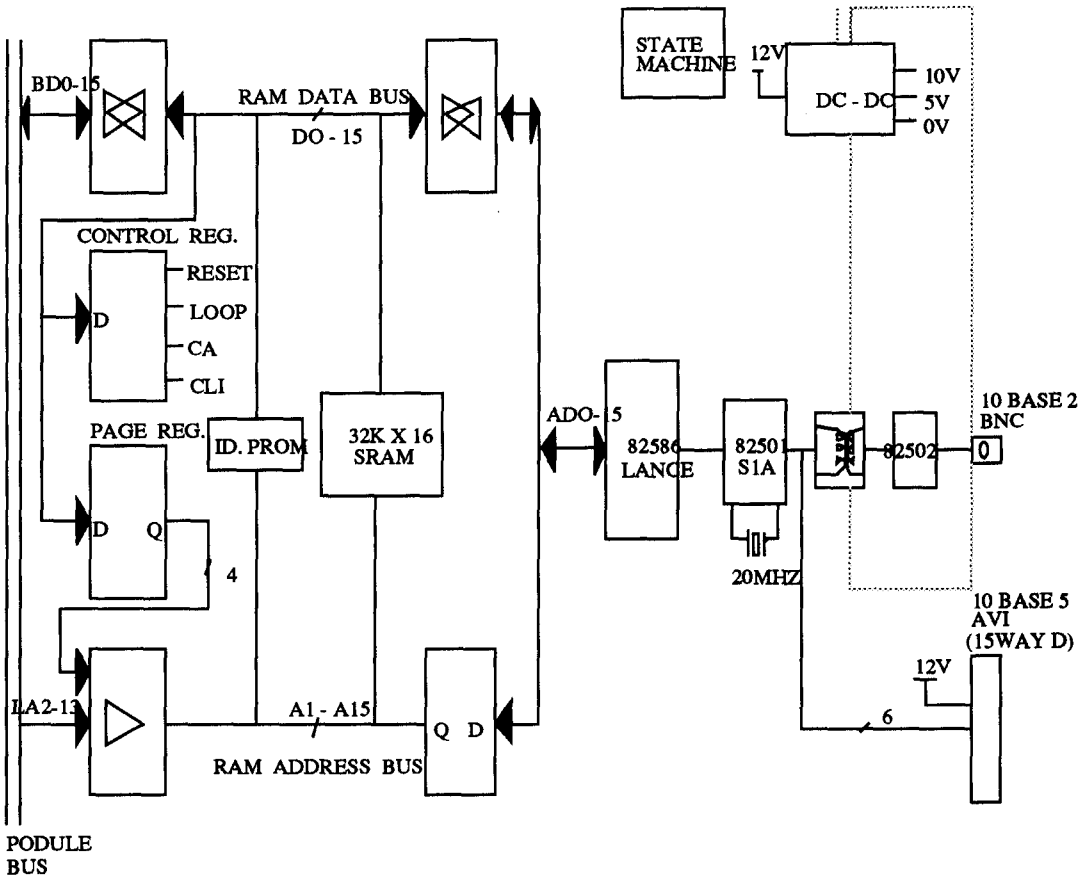


Figure 21.1 : Ethernet podule block diagram

The Intel chip set

As the Xerox and IEEE standards have become widely accepted, a number of systems companies have produced VLSI devices that considerably reduce the design effort required t implement a connection. The most notable of these are by Advanced Micro Devices (AMD) an Intel.

The Intel chip set comprising the 82586 local area network coprocessor, the 82501 ethernet serial interface, and the 82502 ethernet transceiver chip has been used in this design.

The 82586 and other similar local area network controllers are generally referred to by the acronym LANCE, even though this is a trademark of AMD.

The 82586 LANCE performs media access control, framing, pre/postamble generation and stripping, source address generation, CRC checking, and short packet detection. In addition diagnostic functions such as Time Domain Reflectometry (TDR) can be performed.

The 82501 serial interface adapter (SIA) performs Manchester encoding/decoding, receive clock recovery and directly drives the attachment unit interface (AUI) to the cable mounted ethernet transceiver. In addition the 82501 operates a watchdog to prevent continuous transmission (a fault condition), and provides a loop-back test facility. A second source for th device is SEEQ who manufacturer a similar part, the DQ8023A. This part however is not identical and will not perform TDR correctly.

The 82502 transceiver applies transmit data to, and removes receive data from the Cheapernet cable interface. This devices performs a similar function to the cable mounted Ethernet transceiver.

The dual port memory

The LANCE is a true coprocessor and is designed to perform scatter-gather DMA. In common with other LANCE chips the 82586 will utilise a significant bus bandwidth when operating on a net running at 10 Mbps (note: this is not simply the serial data rate divided by the parallel bus width). This bandwidth cannot be provided by the ARM processor over the podule bus and so a dual-port memory system has been implemented.

All communication between the ARM and the LANCE is carried out through command blocks in the dual-port RAM (there are no visible registers in the 82586 LANCE). These command blocks and associated data structures are defined and described in Intel's data sheet, listed in Appendix E.

To issue a command to the LANCE the ARM appends the command to the command block list (CBL) in the dual-port RAM. It then raises the channel attention (CA) signal to the LANCE signalling the presence of the new command. The LANCE responds to CA by reading the command from the CBL and executing as required.

The *LAN Components User's Manual* (reference 9 in the Bibliography) contains a considerably more detailed and comprehensive description of the operation of the LANCE. It is recommended that you obtain a copy, as it contains too much information to include here.

Control register

The control register contains four bits:

Reset (RST) Bit 0.
This bit controls the RESET pin on the LANCE. This bit is set (LANCE reset) on system power-up/hard reset or writing to the control register with this bit logic 1. This bit is cleared (and the LANCE released from the reset state) by writing to the control register with this bit logic 0.

Loop-Back (LB) Bit 1
This bit selects the loop-back mode of 82501 SAI chip. This bit is set and the SIA chip put into loop-back mode by the ARM writing to the control register with this bit logic 1. This bit is cleared (SIA taken out of loop-back mode) on system power-up/hard reset or writing to the control register with this bit logic 0.

Channel Attention (CA) Bit 2
This bit generates a correctly timed CA pulse when the ARM writes to the control register with this bit logic 1. No CA pulse is generated if the ARM writes to the control register with this bit logic 0.

Clear Interrupt (CLI) Bit 3
This bit clears the podule interrupt flag and removes the podule interrupt when the ARM writes to the control register with this bit logic 1. The podule interrupt and flag are unaffected if the ARM writes to the control register with this bit logic 0.

Each bit in the control register is not independent and when writing to a particular bit, the remaining three must be valid. The remaining 12 bits are ignored by the hardware (zero is recommended).

Podule identification PROM

The podule identification PROM contains the following information:

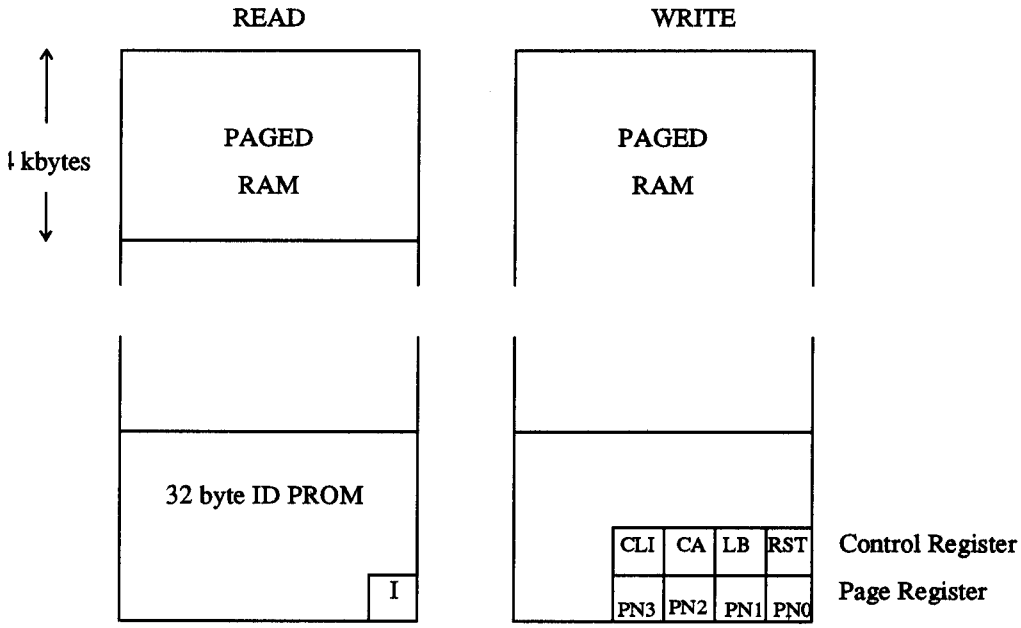
- the Acorn podule identity number (03)
- the interrupt (IRQ) flag bit
- the PCB revision number
- the six byte IEEE globally assigned address block
- a CRC to allow the PROM to be validated.

The contents and operation of the interrupt flag are described in *Interrupts* in *Detailed description* below.

20.5 Detailed description

Address map

The Ethernet podule address map (offset relative to slot base) is shown in the table below. The RAM buffer occupies the upper half of the podule address space. The ID PROM, page register and control register occupy the lower half.



ID PROM (Base)	= Slot base	(Type 3 access) + 0x0000 (Read only)
Page Register	= Slot base	(Type 2 access) + 0x0000 (Write only)
Control Register	= Slot base	(Type 2 access) + 0x0004 (Write only)
Dual-Port RAM	= Slot base	(Type 2 access) + 0x2000 (Read/write)

Table 21.1 : Ethernet podule address map

The LANCE

The 82586 LANCE is a 'scatter-gather' DMA controller type device and is designed to interface to 80186 type processors using a HOLD/HOLDA protocol to resolve arbitration for access to shared memory.

The ARM podule bus cannot easily support a HOLD/HOLDA type interface. This is becaus the ARM is a dynamic device and cannot be stopped for the required time. (This can be long than 10 µs during the interframe/interpacket spacing time.) The ARM cannot be given priori and HOLDA deasserted because this will result in the net controller failing to meet the timin requirements of the net protocol due to the increased bus latency. For example, this could resu in the failure of the net controller to take part in the back-off and retry sequence following a collision on a heavily loaded net.

In this design HOLD and HOLDA are wired together and ARM cycles cause wait-states to b inserted into the LANCE bus cycle. This is achieved by removing the READY signal to the LANCE while the ARM is active. Adopting this scheme avoids the problems outlined above The ARM is never stopped and the LANCE sees minimal bus latency.

The LANCE ARDY/SRDY input used can be programmed to be either asynchronous/ ARD and internally synchronised, or synchronous/SRDY and externally synchronised. In this case is SRDY mode that must be selected. This is achieved by issuing a configure command with th ARDY/SRDY bit set to logic 1. This is important as the LANCE powers-up in ARDY mode

In certain circumstances the LANCE needs to perform read-modify-write bus cycles with lockout. Using READY to insert wait-states does not allow this. However lockout is only required when the LANCE updates error counts (statistics) and even then a problem only aris when a count overflows and the ARM resets it to zero while the LANCE is in the modify phas of a read-modify-write cycle. This is solved by the ARM reading back the count after it sets to zero. If the count is still indicting an overflow then a read modify-write cycle was in progre and the ARM has to correct the count. Error counts this high indicate a major problem that wi require correction so should be a rare event.

The memory bus of the LANCE is operated in 'minimum mode' as the timing parameters fo LANCE outputs in this mode are subject to less spread between devices. The pull-up resisto on WR*, RD*, and BHE are required to prevent RAM cycles when the LANCE is inactive.

The LANCE communicates directly with the SIA (IC24) via a serial channel comprising seve signals: TXC, TXD, RXC, RXD, RTS, CRS and CDT. The function of each of these is described in the LANCE data sheet. The Clear-to-Send (CTS*) input is not supported by the SIA and is connected to 0 V (enabled).

Dual port RAM

The podule bus provides only a limited space in the address map (8 kbytes) for each podule. This is insufficient and so a paged scheme has been implemented.

Viewed from the ARM side the RAMs are paged into the top half of podule space by a 'page register'. The four bit page register is split across two PALs (see the section *The PALs* below Sixteen pages each of 4 kbytes provide 64 kbytes in total. This is organised as 32 k x 16 bits (two 32 k x 8 static RAMs). An alternative RAM size of 8 k x 16 bits (two 8 k x 8 static RAM can be supported (see the section *Links* later in this chapter).

The podule address bus (LA2-13) is buffered by two HCT244 (IC66 and IC58) and the podul data bus (BD0-BD15) is buffered by and two HCT245 transceivers (IC15 and IC54). The direction of the data bus transceivers is determined by the podule R/W signal, while both outpu enables (AAOE and BDOE) are generated by the bus control PAL (IC36).

Viewed from the net controller side, the RAM will be contiguous from location 0x0000 to 0xFFFF. The initialisation root for the controller is 0x0FFFFF6 which is mapped into the RA at 0xFFF6. The high order address bits are not decoded.

The LANCE address/data bus (AD0-AD15) is demultiplexed by two HCT245 (IC17 and IC2 which use the LANCE ALE signal to latch the address bus. The data bus only requires buffe and two HCT573 transceivers (IC10 and IC32) are used. The direction of the data bus transceivers is determined by the LANCE DT/R signal, while the output enables are generate by the bus control PAL (IC36).

The LANCE is capable of operating on an eight bit bus and is reset to this mode. The LANC initialisation root (read when released from reset) contains a bit that defines the bus width an this must be set to 0 (=16 bit bus). Until the LANCE reads this it deasserts Byte High Enabl (BHE*) and outputs address bits on AD8-AD15 for the entire cycle. To avoid a bus clash BHE is used to disable the high order data bus transceiver via the bus control PAL (IC36).

Once initialised to a byte wide bus the LANCE only operates on half words (never bytes) so not necessary to decode the least significant address bit (AD0) to produce separate RAM writ strobes for each byte.

Podule Identification PROM

The device used is a 32 byte PROM 27LS19 (IC14).

A typical content of an ID PROM is shown in Table 21.2 below.

The ID PROM shares address and data bus buffers with the RAM. Viewed from the ARM sid the ID PROM is byte wide and word aligned.

The podule specification defines two bits in the ID byte to be interrupt flags. This design requires only IRQ interrupts so the FIQ flag is always zero. The IRQ flag is generated by connecting the podule interrupt signal to the most significant address pin. The content of the upper half is similar to the lower half but has the IRQ flag bit set, in this way the interrupt fla is multiplexed 'into' the ID byte.

Bytes 09 - 0E are the six byte Ethernet address unique across all Ethernet equipment from manufacturers worldwide.

The CRC (Bytes 1C - 1F) is calculated on the rest of the PROM (Bytes 00 - 1B) using a 32 bi Autodin - II CRC polynomial. This is the same algorithm as the LANCE uses to perform multicast address filtering (see the section *PROM CRC calculation* below). Since each PROM is unique the CRC is used to perform verification.

The output enable is generated by the bus control PAL (IC36).

	D7	D6	D5	D4	D3	D2	D1	D0	NOTES
1F	C	C	C	C	C	C	C	C	CRC on bytes 00 - 1B
1E	C	C	C	C	C	C	C	C	
1D	C	C	C	C	C	C	C	C	
1C	C	C	C	C	C	C	C	C	
1B	Bytes 11 to 1B = 00								
11									
10	0	0	0	0	0	0	0	1	01 - no FIQs, IRQ = 1
0F	0	0	0	0	0	0	0	0	00 - RSVD
0E	I	I	I	I	I	I	I	I	Unique ID
0D	I	I	I	I	I	I	I	I	
0C	I	I	I	I	I	I	I	I	
0B	1	0	1	0	0	1	0	0	A4
0A	0	0	0	0	0	0	0	0	00
09	0	0	0	0	0	0	0	0	00
08	0	0	0	0	0	0	0	1	01 - PCB rev. eg one
07	0	0	0	0	0	0	0	0	00 - UK
06	0	0	0	0	0	0	0	0	Acorn
05	0	0	0	0	0	0	0	0	
04	0	0	0	0	0	0	0	0	Ethernet
03	0	0	0	0	0	0	1	1	
02	0	0	0	0	0	0	0	0	00 - RSVD
01	0	0	0	0	0	0	0	0	00 - no boot code
00	0	0	0	0	0	0	0	0	00 - no FIQs, IRQ = 0

Table 21.2 : Podule identity PROM

The PALs

Three PALs are used in this design:

- the main state PAL (IC29)
- the interrupt and channel attention PAL (IC78)
- the device enable control PAL (IC36).

The main state PAL (IC29)

This PAL implements a state machine which provides timing information for the other two PALS in the design. In addition it produces the two least significant bits of both the page registe (PR0 and PR1) and control register (RSTO and LOOP).

The interrupt and channel attention PAL (IC78)

This PAL implements the two most significant bits of both the page register (PR2 and PR3) an control register (CLI and CA).

The device enable control PAL (IC36).
This device decodes the address map to provide various device output enables.

The state machine and operation

The state machine has four states; IDLE, SA1, SA2, and SA3 and is clocked from state to state on the falling edge of CLK8, the 8 MHz podule bus clock. The figure below is the state diagram

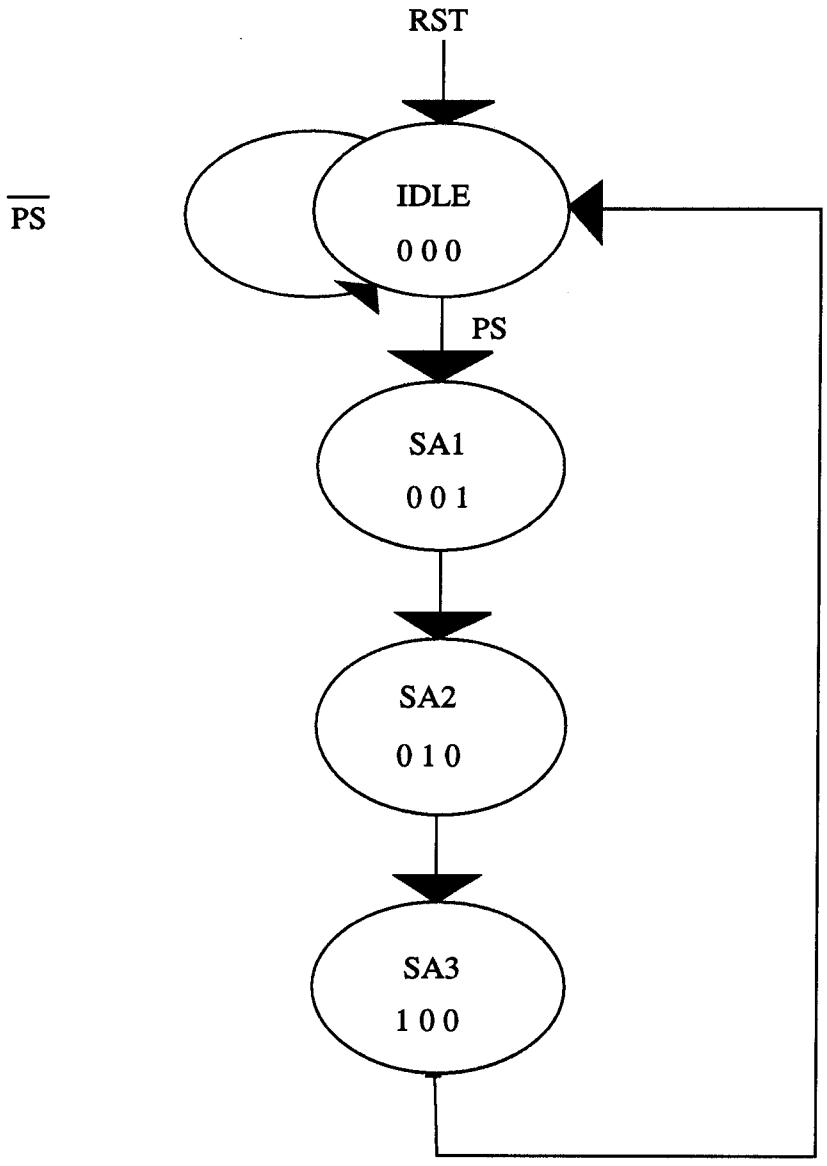


Figure 21.2 : State diagram

The idle state
The state machine enters this state on power-up, hard reset (RST* low), or from the SA3 state. In this state the bus buffers on the ARM side of the dual-ported RAM are disabled and those on the LANCE side enabled. Other outputs such as the page and control register bits remain unchanged. The state machine remains in the idle state until the ARM starts an access (podule select - PS active).

The SA1 state

This state is entered from the idle state only. In this state the LANCE READY signal is disabled, forcing the LANCE to insert wait states if it is active on the bus. The RAM write strobe (RAMWE*) is disabled to prevent writes while the LANCE side of the dual-port RAM is disabled and the ARM side enabled. The state machine exits to the SA2 state unless a reset occurs.

The SA2 state

This state is entered from the SA1 state only. In this state the ARM access is performed and the corresponding device enables are active eg, if a RAM write is performed then the RAM write strobe (RAMWE*) is active. Similarly if a RAM or ID read is required than the RAM or IDOE is active. Writes to the page register or control bits are also performed during this state. READY is still inactive. The state machine exits to the SA3 state unless a reset occurs.

The SA3 state

This state is entered from the SA2 state only. The RAM write strobe (RAMWE*) is disabled to prevent writes while the LANCE side of the dual-port RAM is enabled and the ARM side disabled. The state machine exits to the idle state where any LANCE access that was in progress is completed.

Podule bus cycles

The podule specification requires all ID PROM access to be made using type 3 (sync) IOC bus cycles. All other accesses to the Ethernet podule must be made using type 2 (fast) IOC cycles.

Figure 21.3 below illustrates a read/write to RAM while the net controller is active. The cycle starts with podule select (PS) active and puts the state machine into the SA1 state on the next clock edge. A description of each state that follows is given above.

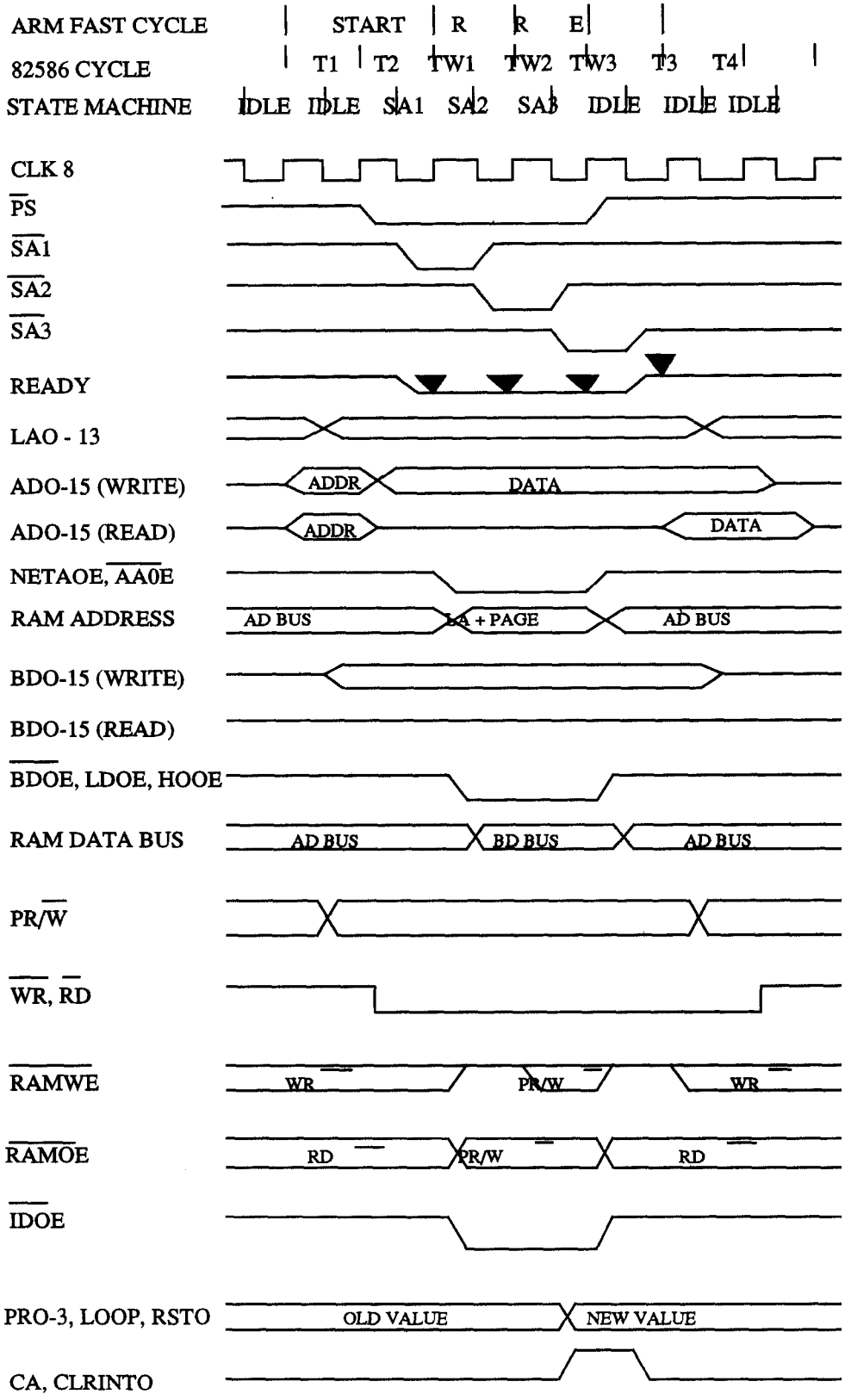


Figure 21.3 : Typical podule bus cycle

It should be noted that *Ready* is always deasserted for three cycles, even if the LANCE is idle. A podule bus access can 'collide' with a LANCE access in five different ways, depending on what state the LANCE is in when the podule bus access starts. These are: PS* while the lance is in states T1 to T4 or idle. The actual number of wait states that the LANCE will insert depends on which of these cases apply. The following four figures illustrate the possible cases.

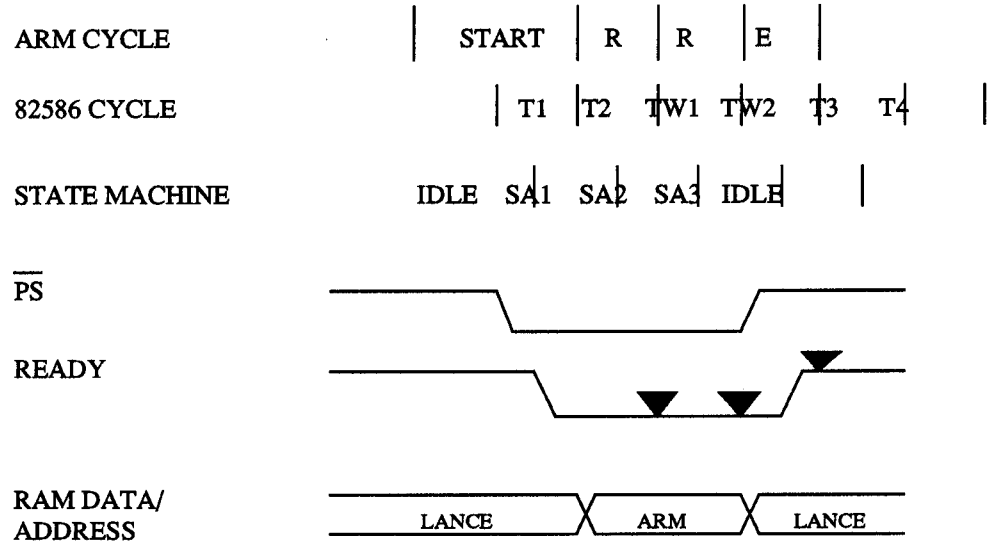


Figure 21.4 : Access collision cases - PS* while LANCE is in T1

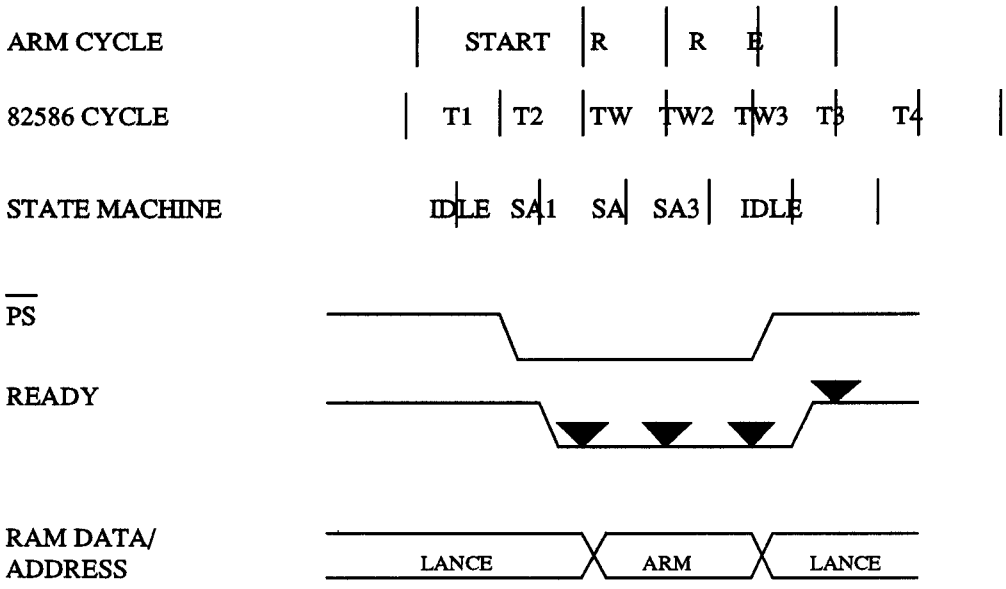


Figure 21.5 : Access collision cases - PS* while LANCE is in T2

PS* while the LANCE is in T1. Figure 21.4.
The LANCE samples READY deasserted at the end of T2 (SA2), and then again at the end of TW1 (SA3), so in this case two wait states are inserted.

PS* while the LANCE is in T.2. Figure 21.5.

The LANCE samples READY deasserted at the end of T2 (SA1), TW1 (SA2), TW2 (SA3), s
the maximum of three wait states are inserted. Since three wait states is the worst case this is
shown in more detail in Figure 21.5.

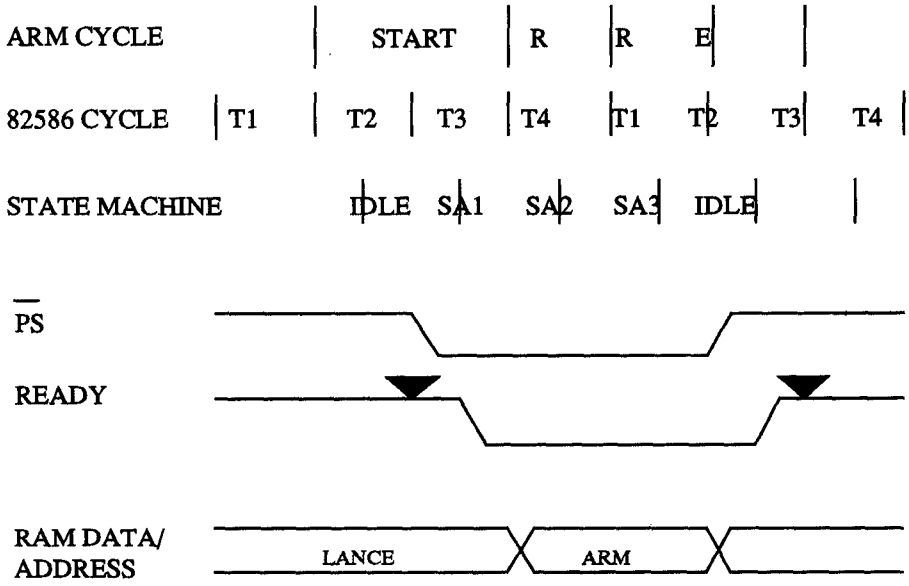


Figure 21.6 : Access collision cases - PS* while LANCE is in T3

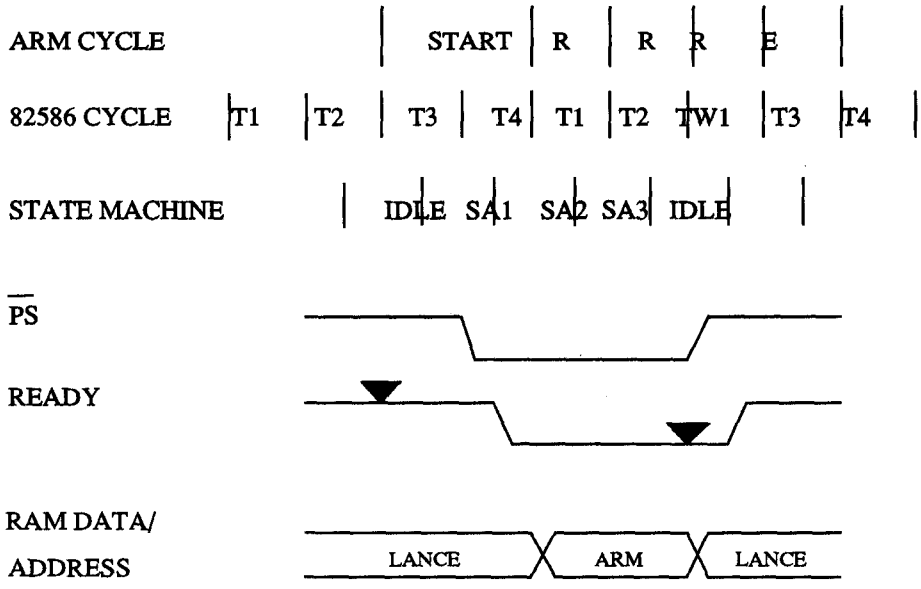


Figure 21.7 : Access collision cases - PS* while LANCE is in T4

PS* while the LANCE is in T3. Figure 21.6.

In this case READY is still active when the LANCE samples it at the end of T3 (idle). This is the last time that the LANCE does this for the current cycle so the LANCE cycle completes before the podule bus cycle starts. Note that the LANCE is not active on the RAM bus during T4.

PS* while the LANCE is in T4. Figure 21.7.
Since the LANCE does not require the bus during T4 no further wait states are inserted in the current cycle. However T1 of the next cycle could follow T4 and one wait state will be inserted into this LANCE access.

PS* while the LANCE is idle.
If the LANCE remains idle while the podule bus cycle occurs then there is no collision and the LANCE ignores the READY signal. This case is not illustrated.

A read from the podule ID PROM or write to the control or page register is similar to a RAM cycle. To simplify the bus design the LANCE is removed from the RAM buses during cycles to these devices.

Bus design note

The cycle stealing scheme should guarantee that the LANCE never has insufficient bus bandwidth or sees excessive bus latency to the extent that it cannot service the net or fails to meet the IEEE timings. Even when the ARM continuously accesses the RAM. The following gives the reasoning behind this statement:

Assumptions:

Net Clock = 10 MHz
Bus Clock = 8 MHz
LANCE FIFO size = 16 bytes

HOLDA is wired to HOLD so:

Bus Latency = 0 cycles
IEEE Interframe Space Time = 9.6 µS

Criteria:

1. FIFO must not over/underrun.

   FIFO fill/empty time from serial side:
   = 8 (bits) * 16 (bytes) * 100E-9 (bit time)
   = 12.8 µs

   FIFO empty/fill time from parallel side:
   = 8 (Word transfers)
   * (4 (standard 8 MHz cycles) + Nwait (wait cycles))
   * 125E-9
   = 4 µs (if Nwait = 0)
   = 7 µs (if Nwait = 3)
   = 8 µs (if Nwait = 4)

2. The LANCE must be in a position to transmit by the end of the interframe spacing time.

   With a Fp/Fs ratio of 8 MHz/10 MHz (0.8):
   16*Nwait + Nlatency must be less than or equal to 80.
   If HOLDA = HOLD then Nlatency = 0
   and
   Nwait <= 5

So this strategy works if we can keep the number of wait states (Nwait) less than or equal to five per access. In the current design three are used and this is unlikely to change.

Interrupts

The podule interrupt (PIRQ) is level triggered. However, the interrupt signal (INT) from the LANCE is designed for use with edge triggered interrupt controllers. If the net controller detects a second interrupting condition just after the first is raised, it will drop and reassert INT. The situation could arise where the podule manager (software) may scan the slots and find no IRQ flag set.

The above problem is prevented by latching INT in the interrupt and channel attention PAL (IC78) and using the latched signal INTO to generate the flag. The clear interrupt (CLI) bit in the control register is used to clear the latch.

Latching INT introduces a further problem which is eliminated by a feature of the 82586 LANCE. If a second interrupt occurs after the processor has read the status word in the SCB but before the first is cleared then the second interrupt would be missed. However, if a the interrupt is cleared at the same time as the channel attention (signalling the acknowledge command) is issued, the LANCE will respond by deasserting INT and reasserting if the second interrupt was not acknowledged because it was missed. It is recommended to set CA whenever CLI is set.

Operation of the interrupt latch and the clear interrupt bit is illustrated in Figure 21.8, Figure 21.9 and Figure 21.10 below.

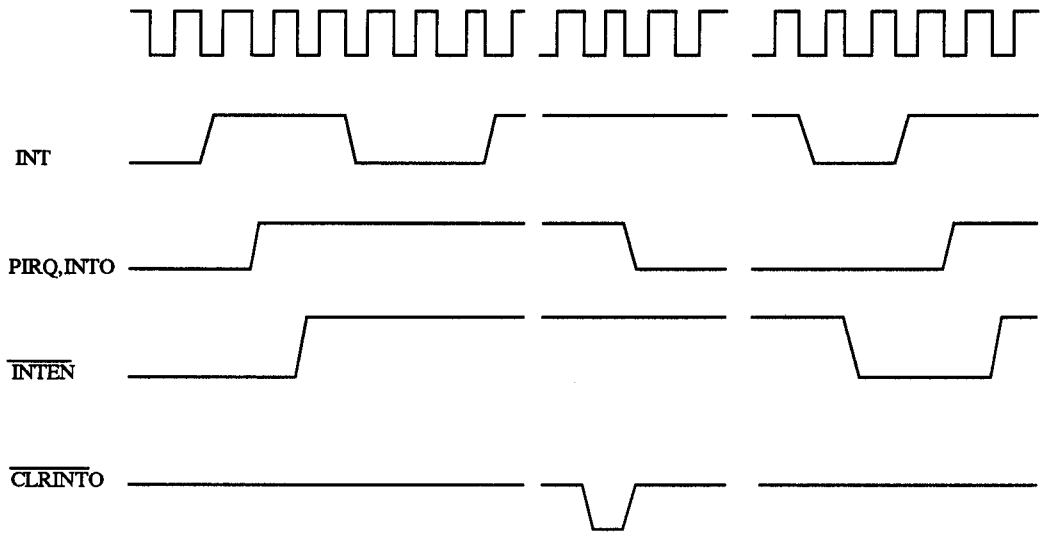


Figure 21.8 : Example interrupt cycles

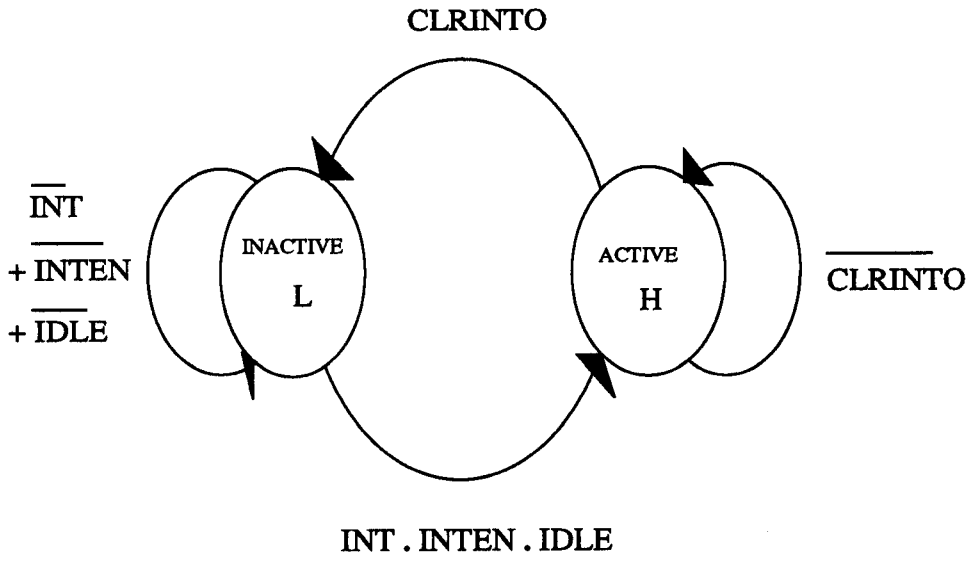


Figure 21.9 : State diagram for INTO/PIRQ*

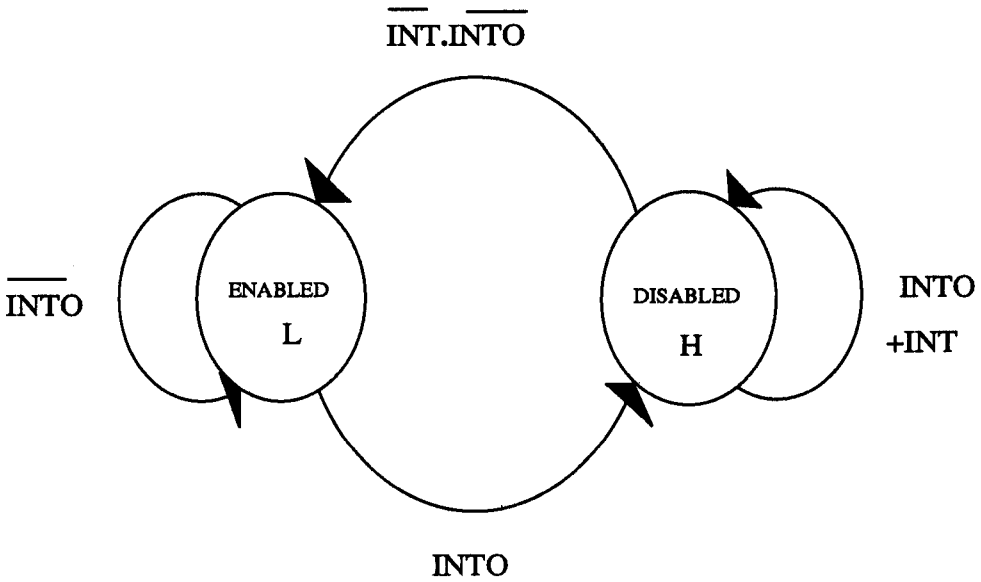


Figure 21.10 : State diagram for INTEN*

Links The PCB should be viewed from the component side with the 96 way podule bus connector on the left and the rear panel on the right. When viewed like this, west is to the left, east the right, north the top and south the bottom.

LK1 and LK2 select the RAM size
If 32 kbyte devices are fitted (normally) the links should both be south. 8 kbyte devices will not normally be fitted but in this case LK1 and LK2 should be north.

LK3 to LK8 select Ethernet or Cheapernet.
For Ethernet operation the links should be west (link pin a to pin b). For Cheapernet operatio the links should be east (link pin b to pin c).

LK9 is tracked south and not fitted on production units.
See data sheets for the 82502 for use.

PROM CRC calculation

The following is a code fragment in the C programming language that calculates and validate the Ethernet PROM checksum.

```
/* To calculate and check the PROM checksum */

int ROM_chk(vector)                          /* array 0..32 bytes    */
u_char vector[32];
{  register int i,j;
  register unsigned chk = -1;                /* Set the CRC register */
                                             /* to FFFFFFFF          */
  register unsigned byte;                    /* temp

  for (i = 0; i < 28; i++) {                 /* CRC on bytes 0..28   */
     byte = vector[i];
     for (j = 0; j < 8; j++) {
        if (((byte & 1) ^ (chk >> 31)) != 0)/* IF feedback = 1      */
           chk = (chk << 1) ^ (0x04C11DB7); /*    shift and EOR taps*/
        else                                 /* ELSE
           chk = (chk << 1);                 /*    just shift        */
        byte = byte >> 1;                    /* next bit             */
     }
  }

/* chk is now the calculated CRC */

/* Now get CRC from PROM */

  byte = (vector[31] << 24) | (vector[30] << 16) | (vector[29] << 8)
         (vector[28] << 0);


/* Test to see if the same */

  if (byte != chk) return (FALSE); /* checksum error*/
  else return (TRUE);
}
```